



9

- (19) The Japanese Patent Office
(12) Laid-Open Patent Application Publication (A)
(11) Laid-Open Patent Application Publication No. Hei4-251297
(43) Publication Date: September 7, 1992

Int. Cl. ⁵	Classification Symbol	JPO Ref. No.	FI
G10H 7/10		L 8946-5H	
G10L 9/02		8622-5H	G10 7/00

Request for Examination: not yet requested

Number of Claims: 7 (11 pages in total) 16

- (21) Application No. Hei3-162514
(22) Application Date: June 7, 1991
(72) Inventor: Tetsuya NIAHIMOTO
c/o Yamaha Corp., No. 10-1, Nakazawa-cho,
Hamamatsu-shi, Shizuoka-ken, Japan
(31) Priority No.: Hei2-410847
(32) Priority Date: December 15, 1990
(33) Priority Country: Japan (JP)
(71) Applicant: Yamaha Corp.
No. 10-1, Nakazawa-cho, Hamamatsu-shi,
Shizuoka-ken, Japan
(74) Agent: Patent Attorney, Yoshihito IIZUKA

(54) Title of the Invention: MUSICAL TONE SYNTHESIZER

[00015]

[Embodiment]

Referring to accompanying drawings, an embodiment of the present invention is described in detail. FIG. 1 is a block diagram, showing a hardware constitution as an embodiment of the electronic musical instrument according to the present invention. In this embodiment, the musical tone synthesizer as a whole is controlled with a microcomputer. The microcomputer includes a microprocessor unit (CPU) 10; a program ROM 11 for storing a system program and various parameters or the like that are not required to be changed; and

a data and working RAM 12 that temporarily stores various data and is used as a working RAM.

[0016]

Preset values of formant sequence are stored in the program ROM 11. Regions for storing the order of the formant sequence, pitch sequence, and voiced and voiceless sound level sequence are assigned to the data and working RAM 12. Thus, the preset values of the formant sequence in the program ROM 11 are loaded by the initialization process starting when power is turned on, so that the musical tone synthesizer starts synthesizing musical tones according to the formant sequence from the time power is turned on.

[0017]

To this microcomputer are connected, through a data and address bus 13, various devices such as a keyboard circuit 14, an operation panel 15, a voiced formant synthesizing sound source 17V and a voiceless formant synthesizing sound source 17U as sound sources, a sequence memory 20, an after-touch sensor 21, and a timer 22. These devices are respectively controlled with the microcomputer.

[0018]

The keyboard circuit 14 includes a circuit made of a plural number of key switches provided so as to correspond to respective keys which specify pitches of musical tones to be produced. Sound producing process for assigning a depressed key to any of a plural number of sound producing channels according to the output from the keyboard circuit 14 is performed with the microcomputer. A process of producing an initial touch data ITD by determining the key depression speed is also performed as required when a key is depressed. An after-touch sensor 21 for outputting after-touch data ATD related to each key of the keyboard by detecting the key depression force when a key is depressed and held is also provided next to the keyboard circuit 14.

[0019]

The operation panel 15 includes various operation members

for choosing, setting and controlling tone color, volume, pitch, and effect. The operation panel 15 has a tone color choosing section 16 for choosing tone colors corresponding to various natural musical instruments such as the piano, organ, violin, brass, guitar, etc. and various other tone colors (human voices). The tone color choosing section 16 outputs tone color choosing signals TC.

[0020]

The voiced formant synthesizing sound source 17V is capable of: simultaneously producing musical tone signals in a plural number (n pieces) of channels; receiving various data given through the data and address bus 13 for keys assigned to respective channels, the data including key code KC data, formant parameter data (center frequency data F and its level data L), relative pitch data P, voiced sound level data V (or level ratio UV of voiced to voiceless sound levels), and other data (such as key-on KON, key-off KOF, initial touch data ITD, after-touch data ATD, and tone color choosing signal TC); and producing musical tone signals according to such data.

[0021]

In other words, the formant synthesizing sound source 17V receives: the center frequency data F, level data L, relative pitch data P, and voiced sound level data V (level ratio UV), of each formant, which have been read from the sequence memory 20; and the key code KC given, as the data specifying the pitch of the musical tone to be synthesized, from the keyboard circuit 14, performs formant synthesizing operation, and outputs a musical tone signal characterized by the formant at a level corresponding to the voiced sound level data V (level ratio UV) corresponding to the pitch specified with the key code KC and the relative pitch data P. The pitch of the musical tone is obtained by multiplying the pitch corresponding to the key code KC by the value of the relative pitch data P. As a matter of course, the pitch may be obtained by other method than multiplication. This embodiment will be described assuming that the number of channels in which the voiced formant

synthesizing sound source 17V can produce sounds simultaneously is 8.

[0022]

A sound source 17U for voiceless formant synthesizing is capable of: simultaneously producing musical tone signals in a plural number (n pieces) of channels; receiving formant parameter data (center frequency data F and its level data L), voiceless sound level data U (or $1 - UV$), and other data (tone color choosing signal TC, etc.) given through the data and address bus 13; and producing musical tone signals according to such data.

[0023]

In other words, the formant synthesizing sound source 17U receives the center frequency data F, level data L, voiceless sound level data U (or $1 - UV$), of each formant read from the sequence memory 20; performs specified formant synthesizing operation, and outputs a musical tone signal characterized by the formant at a level commensurate with the voiceless sound level data U (or $1 - UV$). This embodiment is described on the premise that the number of channels in which the sound source 17U for voiceless formant synthesizing can produce sounds simultaneously is 8, the same as that with the sound source 17V for voiced formant synthesizing.

[0024]

Digital musical signals produced with both of the sound sources 17V and 17U for voiced formant synthesizing are converted to analog musical sound signals through a digital-to-analog (D-A) converters 18V and 18U, and outputted through a sound system 19. The sound system 19 is made up of a speaker, an amplifier, etc. to produce musical sounds commensurate with the analog musical sound signals coming from the D-A converters 18V and 18U.

[0025]

The timer 22 is to give interrupt signals regularly to the microcomputer. With this embodiment, the processes of reading formant parameter data, relative pitch data, and voiced

and voiceless sound level data, to be described later, are implemented with the timer interrupt. The sequence memory 20 has in store various formant data such as the center frequency data F and level data L for a plural number of formants, relative pitch data, voiced and voiceless sound level data, in the sequence corresponding to intended formant change mode.

[0026]

FIG. 2 shows the storage state (example of memory structure) in the sequence memory 20 of the formant parameter data (center frequency data F and level data L), relative pitch data P and voiced and voiceless sound level data V and U. In FIG. 2, the vertical axis corresponds for example to phonemes of human voice identified with sequence numbers X (from 1 to m). The horizontal axis corresponds to time-serial change in: the center frequency and level of the formant characterizing the phoneme identified with the sequence number X of the vertical axis, pitch, voiced sound level, and voiceless sound level, stored in the order of step numbers Y (from 1 to n).

[0027]

In one step of the sequence memory 20 are stored four pieces of formant parameter data FLXYN (with X denoting a sequence number, Y a step number, and N a number indicating any one of four in one step) consisting of center frequency data F and level data L characteristic of the phoneme of the sequence number X. Also stored are, corresponding to the formant parameter data FLXYN, one for each: relative pitch data PXY representing the relative amount of change in the pitch to a reference pitch (identified with a key code KC) at the time the formant is pronounced; voiced sound level data VXY representing the voiced sound level at the time the formant is pronounced; voiceless sound level data UXY (X denoting a sequence number, and Y a step number) representing the level of a voiceless sound. In this case, the value of n of the step number Y in respective sequence numbers X need not necessarily be the same but may be different from one phoneme to another.

[0028]

The sequence memory 20 may store voiced to voiceless sound level ratio data UV in place of the voiced sound level data VXY and the voiceless sound level data UXY. This voiced to voiceless sound level ratio data UV represents the ratio of the voiced sound to the sum of the voiced level and the voiceless level assumed to be 1. Therefore, the voiceless sound level may be easily obtained by subtracting this level UV from 1, as $1 - UV$. Storing the voiced to voiceless sound level ratio data UV in the sequence memory 20 is advantageous in that the amount of data in the sequence memory 20 is greatly reduced.

[0029]

Data read from the step number 1 of the sequence number 1 of the sequence memory 20 are: four pieces of formant parameter data FL111, FL112, FL113, and FL114 respectively made up of center frequency data F and level data L; relative pitch data P11, voiced sound level data V11, and voiceless sound level data U11. Data read out from the step number n are: four pieces of formant data FL1n1, FL1n2, FL1n3, and FL1n4; a piece of relative pitch data P1n, voiced sound level data V1n, and voiceless sound level data U1n. These data are read out in the order of the step number Y.

[0030]

Formant parameter data stored in one step may be adapted to be read either in parallel or in time division. Formant parameter data FL111, FL121,, FL1n1 read in the order of steps are formant sequence data that change in time series, with their center frequencies and levels changing subtly in time series.

[0031]

Relative pitch data P11, P12, , P1n read in the order of steps are pitch sequence data that change in time series and that represent the amounts of change in pitch from the reference pitch. Voiced sound level data V11, V12, , V1n and voiceless sound level data U11, U12, , U1n, read out in the order of steps are voiced and voiceless sound level sequence data that change in time series. Formant parameter data, relative pitch

data, and voiced and voiceless sound level data of respective sequence numbers are constituted similarly.

[0032]

Reading out of the sequence memory 20 is controlled with a sequence number address data X and a step address data Y. The sequence number address data X is to specify the sequence number of a series of sequence formant data and the sequence pitch data to be read from the sequence memory 20. The step address data Y is to specify a step to be currently read out in the sequence specified with the sequence number address data X. As an example, the sequence number address data X is produced in response to an appropriate choosing operation at the operation panel 15, for example in linked motion with the tone color choosing section 16 or in response to a dedicated sequence number choosing operation means. The step address data Y is produced with the CPU 10 and inputted to an address in the sequence memory 20. Producing this step address data will be described later.

[0033]

This embodiment is described as an example on the premise that the formant parameter data are two in number, i.e. the center frequency data F and the level data L, and that one musical tone signal is synthesized from four formants using the voiced formant synthesizing sound source 17V and the voiceless formant synthesizing sound source 17U.

[0034]

Formant parameter data stored in the sequence memory 20 are produced using voice waveform analysis method of conventional description. For example, formant data extracted with a method such as linear prediction analysis (LPC), line spectrum pair analysis (LSP), and complex sine wave model analysis (CSM) are used.

[0035]

Relative pitch data are also extracted at the same time when the formant data are extracted. Extraction of the relative pitch is performed by modified auto correlation or zero

cross-correlation, for example. In other words, when an original voice is to be sampled, a subject is requested to produce a voice of a pitch that is close to a predetermined reference pitch. This voice contains fluctuation component of the pitch. Then, the difference between the reference pitch and the subject pitch is extracted as a relative pitch data. Extracting voiced and voiceless levels is also made similarly to extracting the relative pitch data. Extracting the level ratio UV may be made using Fourier analysis to obtain a ratio of frequency higher than 5 kHz to that below it.

[0036]

The operation panel 15 has a mode switching key (not shown) for choosing one out of a plural number of reading patterns. The CPU 10 produces step address data that change with time to read out formant parameter data, relative pitch data, and voiced and voiceless level data for a plural number of steps according to a reading pattern chosen with the mode switching key. Producing the step address data according to this reading pattern is made according to musical tone producing timing in response to the key-on signal KON from the keyboard circuit 14. The step address data may either be stored in advance in the data and working RAM 12 according to the reading pattern or produced by numerical operation.

[0037]

FIG. 3 is a schematic view, showing one example for values of voiced sound level data V, voiceless sound level data U, and voiced to voiceless sound level ratio UV, stored in the sequence memory 20. FIG. 3(a) shows the voiced sound level data V, FIG. 3(b) the voiceless sound level data U, and FIG. 3(c) the voiced to voiceless sound level ratio UV. In FIGs. 3(a) to 3(c), the horizontal axis represents the step address of the sequence memory 20, and the vertical axis the level values of the voiced and voiceless sound corresponding to the step address.

[0038]

The levels of voiced sound and voiceless sound levels in FIG. 3 respectively correspond to the cases in which human

voices [a], [ka], [sa], and [ta] are synthesized with a musical tone synthesizer. In FIG. 3(a), all the waveforms represent the level V_a of a voiced sound (vowel) [a]. In FIG. 3(b), the first waveform represents the level U_k of a voiceless sound (consonant) [k]; the second, the level U_s of a voiceless sound (consonant) [s]; and the third, the level U_t of a voiceless sound (consonant) [t], respectively. While the magnitude of the voiced sound level V_a in FIG. 3(a) is constant, the magnitudes of the voiceless sound levels U_k , U_s , and U_t in FIG. 3(b) are respectively different. Therefore, the musical tone synthesizer can produce human voices [a], [ka], [sa], and [ta] by reading out the waveforms of FIGS. 3(a) and 3(b) in succession in the order of the step address.

[0039]

FIG. 3(c) shows the voiced to voiceless sound level ratio UV stored in the sequence memory 20. In this example, the ratio of the voiced sound level to the sum of the voiced sound level and the voiceless sound level, assumed to be 1, is stored as the voiced to voiceless sound level ratio in the sequence memory 20. In other words, the voiced to voiceless sound level ratio UV is $V/(U + V)$, where V is the voiced sound level and U is the voiceless sound level. Therefore, the magnitude of the voiceless sound level may be easily calculated as $(1 - UV)$ (indicated with broken line in the figure). However, in case the sum of the voiced sound level and the voiceless sound level is not 1 as shown in FIGS. 3(a) and 3(b), as a matter of course it is preferable to store the voiced sound level V and the voiceless sound level U separately in the sequence memory 20. It is also a matter of course even in case the sum of the voiced sound level and the voiceless sound level is not 1, the ratio between them (V/U) may be simply stored.

[0040]

FIG. 4 shows several example patterns of reading the sequence memory 20. In the reading pattern shown in FIG. 4(a), the step address is increased, in succession, from a specified reference address to a specified maximum address (MAX) and held

there. For example, on the assumption that the reference address is the step number 1 and the maximum address is the step number n , the step address increases in the order of the step numbers 1, 2, 3, ..., n with time, and held at n once it is reached. In case for example the sequence number is 1, formant parameter data $FL11N, FL12N, \dots, FL1nN$; relative pitch data $P11, P12, \dots, P1n$; voiced sound level data $V11, V12, \dots, V1n$; and voiceless sound level data $U11, U12, \dots, U1n$, which numbers correspond to the step numbers respectively, are read out in succession, and finally $FL1nN, P1n, V1n$, and $U1n$ are read out continuously. Incidentally, the term address in the following description is supposed to denote the term step address.

[0041]

According to the reading pattern shown in FIG. 4(b), reading starts from the reference address to a specified loop end address (LOOP END) in specified sequence, thereafter repeating increase in address starting from the loop start address (LOOP START) toward the loop end address (LOOP END).

[0042]

According to the reading pattern shown in FIG. 4(c), reading starts from the reference address to a specified loop end address (LOOP END) in specified sequence, followed by repetition of decrease in address starting from the loop end address (LOOP END) toward the loop start address (LOOP START).

[0043]

According to the reading pattern shown in FIG. 4(d), reading starts from the reference address to a specified loop end address (LOOP END) in specified sequence, followed by repetition of increase and decrease in address between the loop end address (LOOP END) and the loop start address (LOOP START).

[0044]

The reading patterns in FIG. 4 are examples. It is possible to combine these patterns arbitrarily or to variably control the reading speed according to key operation speed information such as the magnitude of the initial touch data ITD and the after-touch data ATD. It is also possible to remarkably

enhance performance effect by pre-registering various patters according to kinds of musical instruments and kinds of human voices.

[0045]

It is also possible to provide a manual operation member on the operation panel 15 to set any step address, or sequentially change the address by changing the amount of manual operation. As such operation members, there are for example a modulation wheel for outputting signals in 128 steps, and a pitch bend wheel for outputting signals having positive and negative directionality.

[0046]

Next, an example process implemented with the microcomputer is described in reference to FIGs. 5, 6, 7, 8, and 9. FIG. 5 shows details of every step of "main process routine" implemented with the microcomputer. This main process routine is implemented sequentially in the following steps.

[0047]

Step 31: All the data of the microcomputer at the time of turning on power are set to specified values. For example, initial values such as the sequence number address data, step address data, and reading pattern are set to respective registers. The preset values of the formant sequence stored in the program ROM 11 as described before are loaded to the data and working RAM 12.

[0048]

Step 32: Key switches in the keyboard circuit 14 are scanned.

Step 33: Presence or absence of a key event is determined from the results of the key scanning in the step 32. A key depression is determined to be a key-on event, and a key release is determined to be a key-off event. When a key event is present (YES), the process goes to the next step 34; when not present (NO), it goes to the step 35.

[0049]

Step 34: Sound production assigning process is implemented according to the kind of the key event in step 33.

Step 35: Operation members of the operation panel 15 are scanned.

Step 36: Presence or absence of panel events with the operation member is determined from the result of panel scanning in the step 35. For example, whether or not a mode switching key has been pressed is determined. In case a panel event is present (YES), the process goes to the next step 37, or in case a panel event is absent (NO), it returns to the step 32.

Step 37: Sound production assigning process is implemented according to the result of the panel event in the step 36.

[0050]

FIG. 6 shows details of steps in the "sound production process" implemented with the microcomputer. This sound production process is implemented in the following order of steps.

Step 41: Whether the key event is in key-on state or key-off state is determined. In case the event is in key-on state (YES), the process goes to the next step 42; in case it is in key-off state (NO), the process goes to the step 47.

Step 42: A vacant channel is searched in which neither the voiced formant synthesizing sound source 17V nor the voiceless formant synthesizing sound source 17U currently implement a sound production assigning process.

[0051]

Step 43: Presence or absence of vacant channel is determined according to the result of the vacant channel search in the step 42. In case it is present (YES), the process jumps to step 45. In case absent (NO), the process goes to the next step 44.

Step 44: When no vacant channel is present, a truncation process is implemented to form a channel that can implement a sound production assigning process.

[0052]

Step 45: Speed of step address reading phase is determined according to the initial touch data ITD, and is set to the speed register SP (ch). This speed register SP (ch) is provided in each channel, with different value set to each channel. As the reading speeds of the formant sequence, pitch sequence, voiced and voiceless level sequence are different by every key depression, an effect is provided that sounds can be separated even when harmonic keys are depressed.

[0053]

Step 46: A key-on flag is set to the vacant channel so that sound production information in interrupt process is controlled, and the process returns. In other words, no sound production assigning process request is made directly in this sound production process.

Step 47: In case the key event in the step 41 results in key-off, a determination is made if the voiced formant synthesizing sound source 17V and the voiceless formant synthesizing sound source 17U have implemented sound production process of the corresponding key, namely if a corresponding key-on is present. If present (YES), the process goes to the step 48; and if not present, the process returns. This is because of the possibility that a depression-processed key is completely halting sound production due to the truncation process in the step 44.

[0054]

Step 48: Key-off information KOF is sent to a channel corresponding to the key-off event.

Step 49: Key-on flag is reset in the channel to which the key-off information KOF was sent.

[0055]

FIG. 7 shows details of steps in the "panel process" implemented with a microcomputer. This panel process is implemented in the following order of steps. Only parts directly related to this invention are shown here.

Step 51: A determination is made if the mode switching key-on the operation panel 15 has been depressed. If depressed (YES),

the process goes to the next step 52; and if not depressed (NO), to the step 56.

[0056]

Step 52: The value of the mode register MOD is increased by 1 (one) and the process goes to the next step 53. In other words, the value of the mode register MODE is increased by 1 (one) every time this mode changing switch is pressed, to choose any of the reading patterns shown in FIG. 4. As a matter of course, the mode changing switch key is not limited to this one but may be any other one that can change its value at will with a slidable or turnable key.

[0057]

Step 53: As the number of reading patterns in FIG. 4 is four, a determination is made if the value of the mode register MODE is "5." If "5," the process goes to the next step 54; otherwise the process jumps to the step 55. It has only to change the value of this step according to the number of the reading patterns.

[0058]

Step 54: When the value of the mode register MODE in the step 53 is a maximum of 5, "1" is stored here in the mode register MODE. Thus, the value of the mode register MODE circulates with the values of 1, 2, 3, and 4 without overflowing as the mode changing switch is depressed.

[0059]

Step 55: Interruption process vector is overwritten according to the value stored in the mode register MODE. This embodiment is adapted to create the waveforms for reading the formant sequence data, pitch sequence data, and voiced and voiceless sound level sequence data by software process and adapted to form respective reading waveforms by different interrupt processes. Therefore, destination of interruption is changed according to the stored value in the mode register MODE.

[0060]

Step 56: When the mode switching key (not shown) on the

operation panel 15 has not been pressed, process is implemented with other keys on the operation panel 15. For example, in case a tone color choosing event occurs, sequence address number change or the like is made according to the tone color.

[0061]

FIG. 8 shows details of the process of reading the formant parameter data, relative pitch data, and voiced and voiceless sound level data implemented with the microcomputer. FIG. 8 shows an interrupt process with the reading pattern shown in FIG. 4(a), or when the stored value of the mode register MODE shown in FIG. 7 is "1." This routine is implemented in the following order of steps every time an interrupt signal is given from the timer 22.

[0062]

Step 61: Interrupt is prohibited to prevent multiple coincident interrupts.

Step 62: Value "1" is set to the channel number register channel.

Step 63: A determination is made if a key-on flag of a channel corresponding to the value stored in the channel number register channel is set. If set (YES), the process goes to the next step 64; and if not set (NO), the process jumps to the step 613 to implement the process of the next channel with the value of the channel number register channel increased by "1."

[0063]

Step 64: The value of the step address register Y (ch) is increased by "1," and further a value $SP(ch) \times SE$, obtained by multiplying the speed register SP (ch) by sensitivity SE, is added. When this sensitivity SE is made "0," the step number Y increases by "1" at a time; when the sensitivity SE is made a positive value, the rate of increase in the step number Y increases, and when the sensitivity SE is made a negative value, the rate of increase in the step number Y decreases.

[0064]

In other words, when the sensitivity SE is "0," formant parameter data and relative pitch data are read from the

sequence memory 20 in the order of step numbers such as FL11N, FL12N, FL13N, When the sensitivity SE is a positive value, reading from the sequence memory 20 is made in skipping order of step numbers such as FL11N, FL13N, FL15N, ..., resulting in higher reading speeds of the formant parameter data, relative pitch data, and/or voiced and voiceless sound level data. When negative in contrast, the formant parameter data, relative pitch data, and/or voiced and voiceless sound level data are read for the same step numbers such as FL11N, FL11N, FL12N, FL12N, FL13N, FL13N, ..., resulting in lower reading speed.

[0065]

Incidentally, it may also be adapted to use only the speed register SP (ch) in place of $SP (ch) \times SE$ so that positive and negative values commensurate with the touch data may be stored in the speed register SP (ch). It may be alternatively adapted to variably control the values of the sensitivity SE and/or speed register SP (ch) according to the values of the initial touch data ITD and/or after-touch data ATD.

[0066]

Step 65: A determination is made if the sequence number of the step address register Y (ch) is greater than the maximum step value MAX. If greater, the process goes to the step 66; and if smaller, the process goes to the step 67.

Step 66: When no step number greater than the maximum step value MAX of the sequence number is present, the maximum step number MAX of the sequence number is stored in the step address register Y (ch).

[0067]

Step 67: A value 1 is stored in the register N to sequentially read four formant parameter data of each step number Y.

Step 68: Voiced sound level data VXY and voiceless sound level data UXY corresponding to the sequence number X and the step number Y are read from the sequence memory 20 and respectively outputted to the voiced formant synthesizing sound source 17V and the voiceless formant synthesizing sound source

17U. This makes it possible to express subtle changes in the levels of voiced sound and voiceless sound. Further, in case the level ratio UV of the voiced sound of FIG. 3(c) is stored in the sequence memory 20, the level ratio UV is outputted to the voiced formant synthesizing sound source 17V. From the level ratio UV, the voiceless sound level ratio (1 - UV) is calculated and then outputted to the voiceless formant synthesizing sound source 17U.

[0068]

Step 69: Relative pitch data PXY corresponding to the sequence number X and the step number Y are read from the sequence memory 20 and then outputted together with the key code KC to the voiced formant synthesizing sound source 17V. This makes it possible to express subtle displacement in pitch. Incidentally, relative pitch data PXY and the key code KC are not taken into the voiceless formant synthesizing sound source 17U.

Step 610: Formant parameter data FLXY1 corresponding to the sequence number X and the step number Y are read from the sequence memory 20 and then outputted to the voiced formant synthesizing sound source 17V and the voiceless formant synthesizing sound source 17U.

Step 611: The value of the register N is increased by "1."

[0069]

Step 612: If the value of the register N is greater than "4" or not is determined. If greater, as the four formant parameter data FLXY1 - FLXY4 of the step number Y have been read, the process goes to the next step 613. If smaller, as the four formant parameter data have not been read yet, the process returns to the step 610 to implement the process until the four formant parameter data are read.

[0070]

Step 613: The value of the channel number register channel is increased by "1."

Step 614: If the value of the channel number register channel is greater than "8" or not is determined. If greater,

the process goes to the next step 615. If smaller, the process returns to the step 63 to apply the same process to the next channel.

Step 615: The interrupt prohibited in the step 61 is allowed to return to the normal process.

[0071]

FIG. 9, like FIG. 8, shows details of the process of reading formant parameter data, relative pitch data, and voiced and voiceless sound level data implemented with the microcomputer. It shows an interrupt process for the case in which the stored value of the mode register MODE shown in FIG. 7 is "2," or for the case of the reading pattern shown in FIG. 4(b). The steps 71 - 74, 77 - 715 are the same as the steps 61 - 64, 67 - 615. Therefore, their explanations are omitted.

[0072]

Step 75: A determination is made if the value of the step address register Y (ch) is greater or not than the loop end address value LE at the sequence number. If greater, the process goes to the step 76. If smaller, the process goes to the step 77.

Step 76: A loop start address value LS at the sequence number is stored in the step address register Y (ch).

[0073]

Through these steps 75 and 76, it is possible to read formant parameter data, relative pitch data, and voiced and voiceless sound level data in succession according to the reading pattern as shown in FIG. 4(b) from the sequence memory 20. Incidentally, reading patterns of FIGs. 4(c) and 4(d) may be easily realized by modifying the flow of FIGs. 8 and 9. Therefore, their explanations are omitted here.

[0074]

The embodiment described above makes it possible to produce musical tones by changing formants in time series and to change musical tone pitch and voiced and voiceless sound levels in time series. There is also an effect at that time to freely control the rate of change in the formant, pitch, and

voiced and voiceless sound levels by changing the reading speed.

[0075]

Further, any type of musical tone synthesizing method may be used in the voiced formant synthesizing sound source 17V. For example, an amplitude modulation (AM) as described in JP-B-Sho-59-19352 or a frequency modulation (FM) as described in JP-B-Sho-62-42515 may be used. FIG. 10 shows an example of the voiced formant synthesizing sound source 17V that synthesizes the voiced formant according to the amplitude modulation (AM) using a window function.

[0076]

A phase generator 81 shown in FIG. 10 produces phase data corresponding to the center frequency by successively adding up the center frequency data F for setting the formant center frequency. Therefore, the successive addition speed is low when the formant center frequency value is small, and the successive addition speed is high when the formant center frequency value is great. When the added-up value overflows, the value returns to the initial value and repeats successive addition. When a reset pulse RS of a specified time duration is given, the sum value is reset to zero, and output is held to zero for only the specified time duration. The sum output of the phase generator 81 is supplied as address data through a selector 85 to a logarithmic sine (log sin) function table 86.

[0077]

A phase generator 82 is of an accumulator structure and takes in basic pitch frequency data f_0 corresponding to the key code KC, and successively adds up the basic pitch frequency data f_0 . Also this phase generator 82, when its value overflows, returns to the initial value and repeats adding action. The phase generator 82 is adapted to output an overflow pulse (for example a most significant bit MSB) to a differentiating circuit 83.

[0078]

The differentiation circuit 83 is made of a mono-stable

multi-vibrator to output, when an overflow pulse rises up, a reset pulse signal RS of a specified time duration to the phase generators 81 and 84. In other words, the differentiating circuit 83 detects the time point when the output value of the phase generator 82 becomes zero and outputs a reset pulse signal RS at the time point. Therefore, the phase data of the formant center frequency produced with the phase generator 81 are reset for a specified period of time at a frequency corresponding to the pitch of the musical tone to be produced according to the reset pulse RS. As a result, amplitude modulation is performed with the formant center frequency used as the carrier frequency and with the musical tone pitch frequency used as the modulation frequency.

[0079]

The phase generator 84 is a circuit for adding up phoneme modulation wave phase constant K supplied from a tone color parameter supply circuit (not shown) synchronously with specified clock pulses. The phase generator 84 is adapted to retain the final value of the added-up sum when the sum overflows. Then, it resets its contents when a reset pulse RS is supplied and starts adding up again from the initial value. The added-up result of the phase generator 84 is supplied as address data through the selector 85 to the logarithmic sine (log sin) function table 86. Here, the phoneme modulation wave phase constant K is set so that the adding up speed of the phase generator 84 is far lower in comparison with the adding up speed of the phase generator 81.

[0080]

The selector 85 chooses output data of the phase generator 81 when action choice signal SEL is supplied, chooses output data of the phase generator 84 when no action choice signal SEL is supplied, and supplies them as address data to the sine function table 86.

[0081]

The sine function table 86 is a table storing sine function data in logarithmic expression for one period (or may be $1/2$

period or $1/4$ period) and is adapted to output sine function values in logarithmic expression commensurate with address data supplied through the selector 85. Therefore, the sine function table 86 outputs sine function values at a rate commensurate with the added up value at the phase generator 81 or 84.

[0082]

A data shifter 87 is a circuit that shifts the output data of the sine function table 86 according to shift amount data S which is a tone color parameter. The shift amount data S is supplied also from the tone color parameter supply circuit (not shown). The data shifter 87 performs shifting action when an action signal SFT is supplied; when the action signal SFT is not supplied, outputs data as they are coming from the sine function table 86. The shift in the data shifter 87 is an action of shifting on more significant side by the value of the shift amount data S.

[0083]

An adder 88, when an action signal ADD1 is supplied, adds up the output data of the data shifter 87 and the output data of a register 89. When no action signal ADD1 is supplied, the data supplied to the adder 88 are outputted as they are from an output terminal. The register 89 is also adapted to store data that have passed intact through the adder 88. In this case, addition with the adder 88 is made with logarithmic data, which means multiplication for antilogarithms.

[0084]

An adder 810, when an action signal ADD2 is supplied to it, adds up the output data of the adder 88 and level-converted level data L. The addition with the adder 810 is made with logarithmic data, which means multiplication for antilogarithms.

[0085]

A logarithmic-linear (log-linear) conversion circuit 811 is a circuit that converts data supplied from the adder 810 in logarithmic expression into antilogarithms. Data outputted from the logarithmic-linear conversion circuit 811 are given

to an accumulator 812. Four formant parameter data F and L for synthesizing a musical tone are given in time division, and phoneme signals corresponding to respective formants are outputted in succession from the logarithmic-linear conversion circuit 811, added up in the accumulator 812, and outputted as a musical tone to a multiplier 813. The multiplier 813 receives voiced sound level data V (or UV), multiplies it by a musical tone signal coming from the accumulator 812, and outputs the result as a musical tone signal of a voiced sound.

[0086]

The above description is on an example of the voiced formant synthesizing sound source 17V. As detailed description on the voiced formant synthesizing sound source 17V appears in the specification of Japanese patent application No. Hei-1-77383, it is omitted here. Additionally, in the voiced formant synthesizing sound source 17V in FIG. 10, the phase generator 84, the selector 85, and the data shifter 87 may be omitted.

[0087]

FIG. 11 shows an example of the voiceless formant synthesizing sound source 17U. As details of this voiceless formant synthesizing sound source 17U appear in the specification of Japanese patent application No. Hei-2-271397, simplified description is given here. In FIG. 11, a white noise generating circuit 91 is to generate white noise having a flat spectrum. A digital filter 92 is a low-pass filter called IIR filter to convert flat spectrum white noise into noise having a specified band width. The digital filter 92 is made up of: an inverter 93, a band width parameter generator 94, a delay circuit 95, adders 96, 97, 98, and a multiplier 99. The digital filter 92 converts white noise into noise signal having a spectrum characteristic sloping down toward the right and outputs it to a multiplier 910.

[0088]

A periodic waveform generating circuit 911 outputs a sinusoidal sequential sample point amplitude value $\sin 2\pi f_0 t$

having a formant center frequency f_0 transitioning according to center frequency data F for setting formant center frequency. The periodic waveform generating circuit 911 is made up of a phase accumulator 912 and a sine table 913. The phase accumulator 912 adds up center frequency data F synchronously with specified clock pulses. As the center frequency data F correspond to the formant center frequency f_0 of noise sound intended to be produced, the phase accumulator 912 outputs its added-up value as a reading address signal of the sine table 913.

[0089]

The sine table 913 is a table having stored sine function data for one period (or may be $1/2$ or $1/4$ period) to be read out with reading address signals. Therefore, from the sine table 913 comes out a sine wave of a frequency f_0 corresponding to the reading address signal (added-up value) of the phase accumulator 912. The multiplier 910 multiplies the noise signal of the digital filter 92 by the sine wave of the periodic waveform generating circuit 911 and outputs the product. Therefore, noise signal having specified formant characteristic comes out of the multiplier 910.

[0090]

An envelope generator 914 outputs envelope signals, for controlling the amplitude of noise signals outputted from the multiplier 910, synchronously with clock pulses and according to the level data L of the formant parameter, to a multiplier 915. The multiplier 915 multiplies the envelope signal by the noise signal from the multiplier 910 and outputs the product to a next-step multiplier 916. The multiplier 916 receives the voiceless sound level data U (or $1 - UV$), multiplies it by the noise signal from the multiplier 915, and outputs the product as the musical sound signal of the voiceless sound.

[0091]

While the above embodiment is described as realized with software, embodying the invention is not limited to the above but may be realized with hardware. It may also be adapted that

specific values of formant parameter data in the sequence memory 20 can be arbitrarily overwritten by operation on the operation panel 13.

[0092]

The above embodiment is described on the assumption that the step number of FIG. 2 is read according to the reading pattern of FIG. 4. The manner of reading is not limited to the above. That is, formant parameter data may be read out sequentially according to the reading pattern of FIG. 4, moving from the step number to the sequence number. Additionally, while the above embodiment is described assuming that all of the formant parameter data, relative pitch data, and voiced and voiceless sound level data are read, it may be assumed that any one kind of data are read.

[0093]

Amounts of change in the pitch and the voiced and voiceless sound levels are small in comparison with the formant parameter data, and the speed of change is also low. Therefore, relative pitch data and voiced and voiceless sound levels need not be provided for every step number. Instead, it may be adapted to provide relative pitch data and voiced and voiceless sound levels at intervals of several steps, or to store formant parameter data, relative pitch data, and voiced and voiceless sound levels in different sequence memories to be read at different sequence speeds, or to provide different step numbers.

[0094]

Relative pitch data used may be extracted by analysis, or produced with a dedicated editor or the like. A switch may be provided to bring the output of the relative pitch data or voiced and voiceless sound level data to an off state; or a switch may be provided to vary the depth (magnitude) of the relative pitch data or voiced and voiceless sound level data. This is in preparation for a possible need of producing a flat sound without any change both in pitch and in voiced and voiceless sound levels. As for the formant parameter data, like the pitch

data, it may be adapted that the amount of change relative to a certain reference formant data are stored sequentially in time series.

[0095]

While the above embodiment is described assuming that parameters of the center frequency and level for specifying formant are stored as formant parameter data in the sequence memory 20, in case the formant is synthesized by frequency modulation operation, it may be adapted to have in store various parameters for that purpose such as the center frequency, modulation frequency, modulation index, and level, to be read out. As a matter of course, this applies to synthesizing formants of rhythm sounds or the like as well as scale sounds.

[0096]

[Effects of the Invention]

According to the invention, formant parameter data, relative pitch data, or voiced and voiceless sound level data, changing in time series, are pre-stored over a plural number of steps in a memory means. The formant parameter data, relative pitch data, or voiced and voiceless sound level data are sequentially read with a reading means over a plural number of steps. This makes it possible to naturally change the formant, pitch or voiced and voiceless sound levels of musical tones like actual musical instrument sound or human voice.

[Brief Description of Drawings]

FIG. 1 is a block diagram, showing a hardware constitution as an embodiment of the electronic musical instrument according to the present invention.

FIG. 2 shows the state of the formant parameter data and relative pitch data stored in the sequence memory shown in FIG. 1.

FIG. 3 schematically shows an example of values of voiced sound level data and voiceless sound level data stored in the sequence memory.

FIG. 4 shows several example patterns of reading the sequence memory shown in FIG. 1.

FIG. 5 shows a flowchart of an exemplary main routine implemented with the microcomputer shown in FIG. 1.

FIG. 6 shows a flowchart of detailed sound production process of FIG. 4 implemented with the microcomputer of FIG. 1.

FIG. 7 shows detailed example of panel process of FIG. 4 implemented with the microcomputer of FIG. 1.

FIG. 8 shows a flowchart of detailed process of reading formant parameter data, relative pitch data, and voiced and voiceless sound level data according to the reading pattern of FIG. 3(a).

FIG. 9 shows a flowchart of detailed process of reading formant parameter data, relative pitch data, and voiced and voiceless sound level data according to the reading pattern of FIG. 3(b).

FIG. 10 shows an example of voiced formant synthesizing sound source of FIG. 1.

FIG. 11 shows an example of voiceless formant synthesizing sound source of FIG. 1.

[Description of Reference Numerals and Symbols]

- 10: CPU
- 11: program ROM
- 12: data and working RAM
- 13: data and address bus
- 14: keyboard circuit
- 15: operation panel
- 16: tone color choosing section
- 17V: voiced formant synthesizing sound source
- 17U: voiceless formant synthesizing sound source
- 18: D-A converter
- 19: sound system
- 20: sequence memory
- 21: after-touch sensor
- 22: timer

FIG. 1

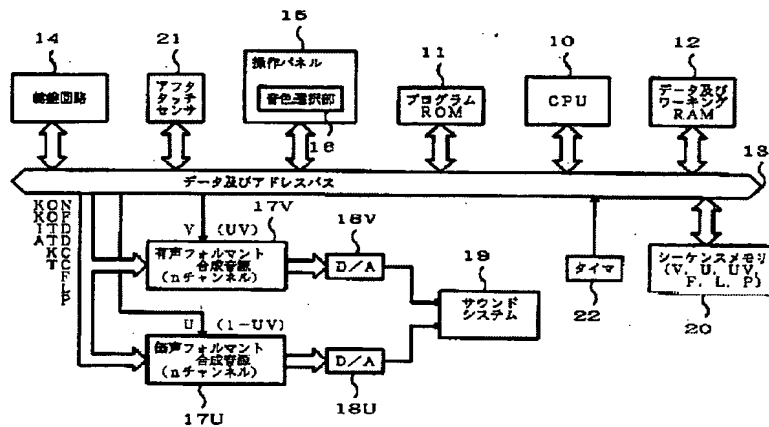


FIG. 5

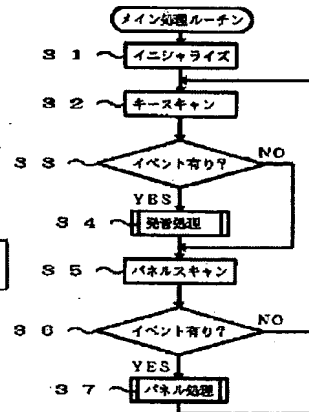


Fig. 1

- 10: CPU
- 11: Program ROM
- 12: Data & working RAM
- 13: Data & address bus
- 14: Keyboard circuit
- 15: Operation panel
- 16: Tone color choosing section
- 17U: Voiceless formant synthesizing section (n channels)
- 17V: Voiced formant synthesizing section (n channels)
- 18U: D/A
- 18V: D/A
- 19: Sound system
- 20: Sequence memory
- 21: After-touch sensor
- 22: Timer

Fig. 5

- Main process routine
- S1: Initialize
- S2: Key scanning
- S3: Is event present?

S4: Sound production process
 S5: Panel scanning
 S6: Is event present?
 S7: Panel process

FIG. 3

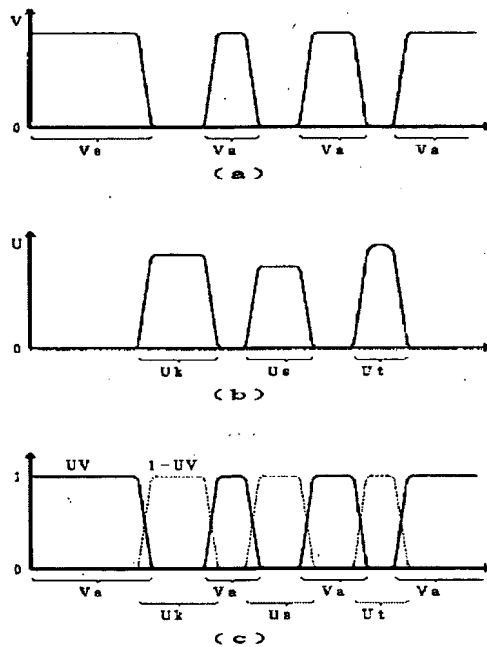


FIG. 6

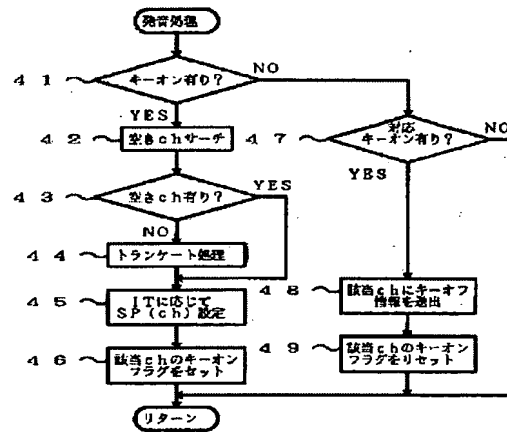


Fig. 6

Sound production process

41: Is key-on present?

42: Vacant channel search

43: Is vacant channel present?

44: Truncate process

45: Set SP(ch) according to IT

46: Set key-on plug of corresponding channel

47: Is corresponding key-on present?

48: Send key-off information to corresponding channel

49: Reset key-on plug of corresponding channel

Return

FIG. 4

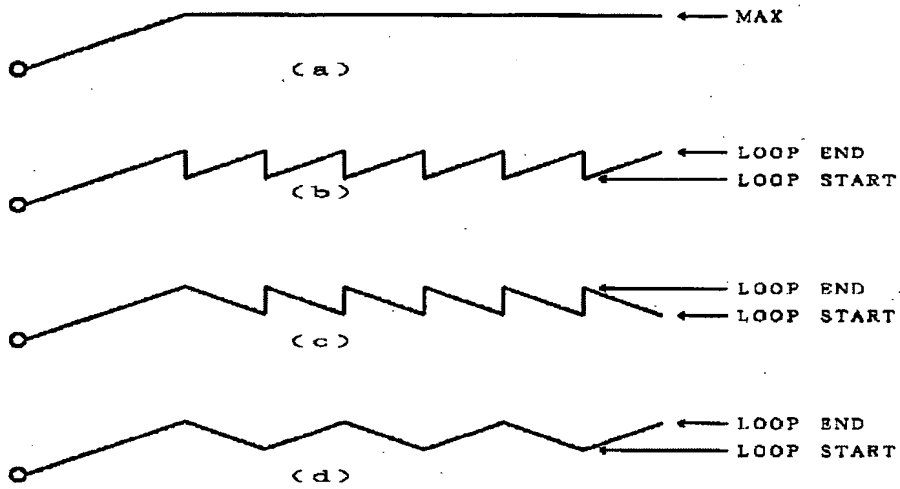


FIG. 7

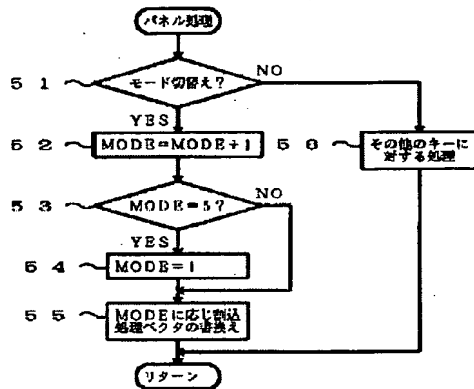


FIG. 8

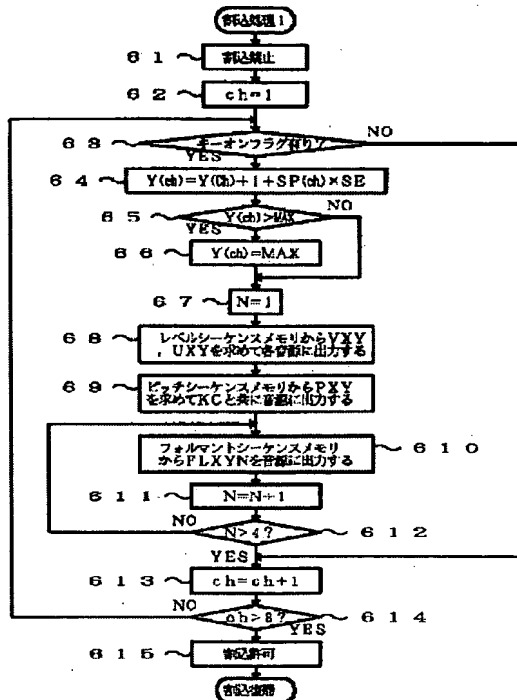


Fig. 7

Panel process

51: Mode switched?

52: MODE=MODE+1

53: MODE=5?

54: MODE=1

55: Rewrite interrupt process vector according to MODE

56: Process to other keys

Return

Fig. 8

Interrupt process 1

61: Interrupt prohibit

62: ch=1

63: Is key-on flag present?

64: $Y(ch) = Y(ch) + 1 + SP(ch) \times SE$ 65: $Y(ch) > MAX$

66: $Y(ch) = MAX$

67: $N=1$

68: Obtain VXY and UXY from level sequence memory, output them to each sound generator

69: Obtain PXY from pitch sequence memory, output it together with KC to sound generator

610: Output FLXYN from formant sequence memory to sound generator

611: $N=N+1$

612: $N>4?$

613: $ch=ch+1$

614: $ch>8?$

615: Allow interrupt

Return interrupt

FIG. 9

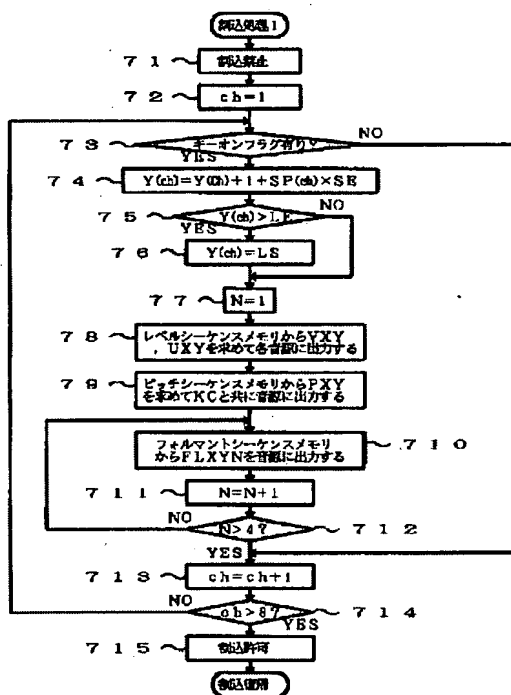


FIG. 10

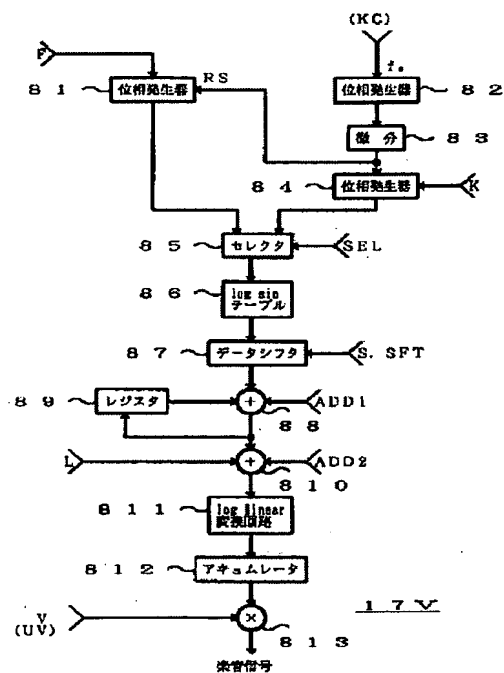


Fig. 9

Interrupt process 1

71: Interrupt prohibit

72: $ch=1$

73: Is key-on flag present?

74: $Y(ch)=Y(ch)+1+SP(ch)*SE$

75: $Y(ch)>LE$

76: $Y(ch)=LS$

77: $N=1$

78: Obtain VXY and UXY from level sequence memory, output them to each sound generator

79: Obtain PXY from pitch sequence memory, output it together with KC to sound generator

710: Output FLXYN from formant sequence memory to sound generator

711: $N=N+1$

712: $N>4?$

713: $ch=ch+1$

714: $ch>8?$

715: Allow interrupt

Return interrupt

Fig. 10

81: Phase generator

82: Phase generator

83: Differentiation

84: Phase generator

85: Selector

86: Log sin table

87: Data shifter

89: Register

811: Log linear conversion circuit

812: Accumulator

Musical tone signal

FIG. 11

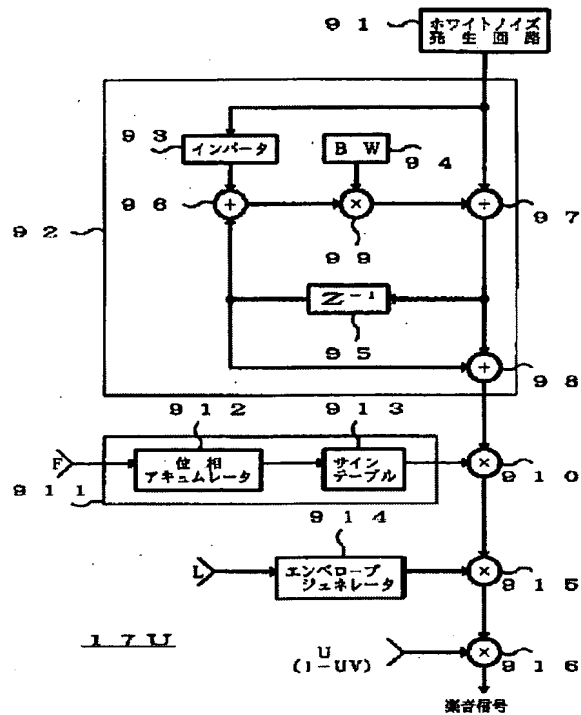


Fig. 11

91: White noise producing circuit

93: Inverter

912: Phase accumulator

913: Sine table

914: Envelope generator

Musical tone signal

(11)特許出願公開番号

特開平4-251297

(43)公開日 平成4年(1992)9月7日

(51)Int.Cl. ⁵	識別記号	序内整理番号	F I	技術表示箇所
G 1 0 H 7/10				
G 1 0 L 9/02	L	8946-5H		
		8622-5H	G 1 0 H 7/00	

審査請求 未請求 請求項の数 7 (全 16 頁)

(21) 出願番号 特願平3-162514

(22) 出願日 平成3年(1991)6月7日

(31) 優先権主張番号 特願平2-410847

(32) 優先日 平2(1990)12月15日

(33) 優先権主張国 日本(JP)

(71)出願人 000004075
ヤマハ株式会社
静岡県浜松市中沢町10番1号

(72)発明者 西元 哲夫
静岡県浜松市中沢町10番1号 ヤマハ株式
会社内

(74)代理人 弁理士 飯塚 義仁

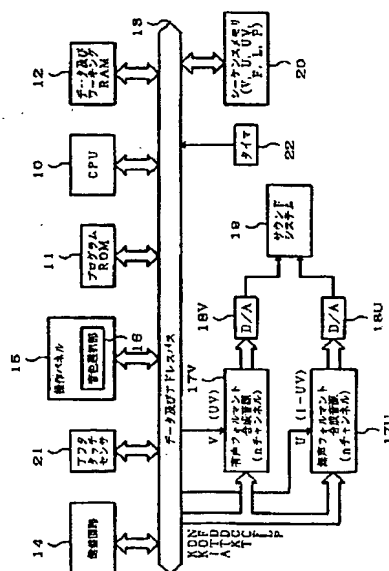
(54)【発明の名称】 楽音合成装置

(57) 【要約】

【目的】自然楽器の楽音や人声音等のフォルマント、ピッチ及び有声無声音レベルの微妙な時間的変化を楽音合成装置で忠実に再現できるようにする。

【構成】自然楽器の楽音や人声音等の時系列的に変化するフォルマント、ピッチ及び有声無声音レベルに関するパラメータデータを複数ステップにわたって予めメモリ上に記憶する。楽音を発生すべきときに、そのメモリからパラメータデータを複数ステップにわたって時系列的に読み出し、読み出されたパラメータデータに応じて決定されるフォルマント特性、ピッチ特性又は有声無声音レベル特性をそれぞれ有する楽音信号を楽音合成装置で合成する。

【効果】楽音信号のフォルマント、ピッチ又は有聲無声音レベルを自然楽器の楽音や人声音等と同様に時間的に微妙に変化させることができる。



(2)

特開平4-251297

1

【特許請求の範囲】

【請求項1】 時系列的に変化するフォルマントに関するパラメータデータを複数ステップにわたって予め記憶している記憶手段と、楽音を発生すべきときに、前記記憶手段から前記パラメータデータを前記複数ステップにわたって時系列的に読み出す読み出し手段と、読み出された前記パラメータデータを入力し、前記パラメータデータに応じて決定されるフォルマント特性を持つ楽音信号を合成するフォルマント合成手段とを備え、前記楽音信号のフォルマントを時系列的に変化させることを特徴とする楽音合成装置。

【請求項2】 基準ピッチに対して時系列的に変化する相対ピッチデータを複数ステップにわたって予め記憶している記憶手段と、楽音を発生すべきときに、前記記憶手段から前記相対ピッチデータを前記複数ステップにわたって時系列的に読み出す読み出し手段と、所定のフォルマント特性に従った音高を有する楽音信号を合成するものであり、前記読み出し手段によって読み出された前記相対ピッチデータを入力し、この相対ピッチデータに応じて前記楽音信号の音高を時系列的に変化させる楽音合成手段とを備えたことを特徴とする楽音合成装置。

【請求項3】 時系列的に変化する有声音及び無声音のレベルデータを複数ステップにわたって予め記憶している記憶手段と、楽音を発生すべきときに、前記記憶手段から前記レベルデータを前記複数ステップにわたって時系列的に読み出す読み出し手段と、所定のフォルマント特性に従った有声音及び無声音の特徴を有する楽音信号を合成するものであり、前記読み出し手段によって読み出された前記レベルデータを入力し、このレベルデータに応じて前記楽音信号の有声音及び無声音のレベルを時系列的に変化させる楽音合成手段とを備えたことを特徴とする楽音合成装置。

【請求項4】 時系列的に変化する有声音及び無声音のレベルデータと基準ピッチに対して時系列的に変化する相対ピッチデータとの少なくとも一方のデータを時系列的に変化するフォルマントに関するパラメータデータと共に複数ステップにわたって予め記憶している記憶手段と、楽音を発生すべきときに、前記レベルデータ及び前記相対ピッチデータの少なくとも一方と前記パラメータデータとを前記記憶手段から前記複数ステップにわたって時系列的に読み出す読み出し手段と、読み出された前記レベルデータ及び前記相対ピッチデータの少なくとも一方と前記パラメータデータとを入力し、入力された前記レベルデータ及び前記相対ピッチデータの少なくとも一方と前記パラメータデータに応じて決定される有声音無声音レベル及びピッチの少なくとも一方とフォルマント特性とを有する楽音信号を合成するフォルマント合成手段とを備え、前記楽音信号の有声音無声音レベル及びピッチの少なくとも一方とフォルマントとを時系列的に変化させることを特徴とする楽音合成装置。

2

【請求項5】 前記読み出し手段は、キー操作情報に応じた読み出し速度で前記記憶手段の読み出しを行うことを特徴とする請求項1、2、3又は4に記載の楽音合成装置。

【請求項6】 前記レベルデータは時系列的に変化する有声音及び無声音のレベルをそれぞれ別々に示すものであることを特徴とする請求項3又は4に記載の楽音合成装置。

【請求項7】 前記レベルデータは時系列的に変化する有声音及び無声音のレベル比を示すものであることを特徴とする請求項3又は4に記載の楽音合成装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】 この発明は、所望のフォルマントに従う楽音合成を実現する楽音合成装置に関する。

【0002】

【従来の技術】 一般に自然楽器には、その楽器固有の構造（例えばピアノの響板の形状等）によって生じるフォルマントが存在することが知られている。また、人声にも人体の構造（例えば声帯、声道及び口腔の形状等）によって所定のフォルマントが存在し、これによって人声特有の音色等が特徴づけられている。電子楽器において、自然の楽器音又は人声音により近い音色を合成するためには、それぞれの音に固有のフォルマントに従って楽音合成を行わなければならない。このようなフォルマントによって楽器音や人声等の楽音を合成する装置として特公昭59-19352号公報に示されたものが知られている。

【0003】

【発明が解決しようとする課題】 上述のように自然楽器や人声の音色は、その楽器に固有のフォルマントによって特徴付けられている。ところで、自然楽器の演奏時には、その発生音のフォルマントが時間的に変化しており、このようなフォルマントの時間的な変化によってその楽器固有の音色が奏でられる。また、人声においても同様であり、フォルマントが時間的に変化することにより、微妙な人声の特徴が出される。

【0004】 フォルマントを時間的に変化させて、このような楽音及び人声音等を電子楽器で合成する場合に、フォルマントを特定するための中心周波数やレベル等のパラメータを時間的に変化させればよいわけであるが、一般的にパラメータの時間的変化を実現する手法としては、適宜のエンベロープ信号によって連続的にこれを変化させるということが行われている。しかし、このような単純な方法では望みのフォルマント変化を得ることが困難であった。

【0005】 一方、人声音は、発音時にピッチがゆらいたり、音の立上り時にピッチが不安定になったり、音楽から音楽に移り変わるときにピッチが変化するなどの独特のピッチ変化を有する。しかし、従来の楽音合成装置

3

は鍵盤等からのピッチ情報に基づいたピッチを単に発音するか、フォルマントの時間的変化を実現する手法と同様にピッチエンベロープ信号発生回路やピッチゆらぎ回路によって基準ピッチを連続的に変化させるだけであった。従って、楽音合成装置でフォルマントに基づいて人声音を発音してもそれは実際の人声音の近似であり、実際の人声音とは程遠い自然性に全く欠けるものであった。

【0006】また、人声音は、有声フォルマント（母音）及び無声フォルマント（子音）からなり、音素から音素に移り変わるときに有声フォルマント及び無声フォルマントのそれぞれのレベルが微妙に変化している。しかし、従来の楽音合成装置はこのような有声フォルマント及び無声フォルマントを人声音に応じて交互に発音処理するだけであり、発音された人声音は実際の人声音とは程遠い自然性に全く欠ける機械的なものであった。自然楽器の楽音も、線スペクトル成分と非線スペクトル成分とからなり、そのレベルがそれぞれ微妙に変化しているのだが、従来の楽音合成装置は、このようなスペクトル成分のレベル制御を行っていなかった。

【0007】この発明は上述の点に鑑みてなされたものであり、自然楽器の楽音や人声音等の時間的変化を忠実に再現することのできる楽音合成装置を提供することを目的とする。

【0008】

【課題を解決するための手段】第1の発明に係る楽音合成装置は、時系列的に変化するフォルマントに関するパラメータデータを複数ステップにわたって予め記憶している記憶手段と、楽音を発生すべきときに、前記記憶手段から前記パラメータデータを前記複数ステップにわたって時系列的に読み出す読み出し手段と、読み出された前記パラメータデータを入力し、前記パラメータデータに応じて決定されるフォルマント特性を持つ楽音信号を合成するフォルマント合成手段とを備え、前記楽音信号のフォルマントを時系列的に変化させることを特徴とするものである。

【0009】第2の発明に係る楽音合成装置は、基準ピッチに対して時系列的に変化する相対ピッチデータを複数ステップにわたって予め記憶している記憶手段と、楽音を発生すべきときに、前記記憶手段から前記相対ピッチデータを前記複数ステップにわたって時系列的に読み出す読み出し手段と、所定のフォルマント特性に従った音高を有する楽音信号を合成するものであり、前記読み出し手段によって読み出された前記相対ピッチデータを入力し、この相対ピッチデータに応じて前記楽音信号の音高を時系列的に変化させる楽音合成手段とを備えたことを特徴とするものである。

【0010】第3の発明に係る楽音合成装置は、時系列的に変化する有声及び無声音のレベルデータを複数ス

(3)

特開平4-251297

4

発生すべきときに、前記記憶手段から前記レベルデータを前記複数ステップにわたって時系列的に読み出す読み出し手段と、所定のフォルマント特性に従った有声音及び無声音の特徴を有する楽音信号を合成するものであり、前記読み出し手段によって読み出された前記レベルデータを入力し、このレベルデータに応じて前記楽音信号の有声音及び無声音のレベルを時系列的に変化させる楽音合成手段とを備えたことを特徴とするものである。

【0011】

【作用】第1の発明では、記憶手段は時系列的に変化するフォルマントに関するパラメータデータを複数ステップにわたって記憶し、読み出し手段は楽音を発生すべきときに、記憶手段からパラメータデータを複数ステップにわたって時系列的に読み出し、フォルマント合成手段に供給し、フォルマント合成手段はそのパラメータデータに応じて決定されるフォルマント特性を持つ楽音信号を合成している。従って、フォルマント特性が自然の楽器音又は人声音に従って変化するようにパラメータデータを複数ステップにわたって予め記憶手段に記憶し、このパラメータデータを読み出し手段で複数ステップにわたって読み出すことによって、楽音信号のフォルマントを実際の楽器音又は人声音と同様に自然に変化させることができる。このパラメータデータは主にフォルマント中心周波数データとそのフォルマントレベルデータとから構成される。

【0012】第2の発明では、基準ピッチに対して時系列的に変化する相対ピッチデータを前記第1の発明のパラメータデータと同様に記憶し、それを時系列的に読み出して楽音信号を合成しているので、楽音信号のピッチを実際の楽器音又は人声音と同様に自然に変化させることができる。

【0013】第3の発明では、時系列的に変化する有声音及び無声音のレベルデータを前記第1の発明のパラメータデータと同様に記憶し、それを時系列的に読み出して楽音信号を合成しているので、楽音信号の有声音及び無声音のレベルを実際の楽器音又は人声音と同様に自然に変化させることができる。

【0014】なお、第2の発明の相対ピッチデータ及び第3の発明の有声無声音レベルデータの少なくとも一方を第1の発明のパラメータデータと共に記憶し、それを時系列的に読み出して楽音信号を合成することによって、楽音信号をより自然の楽器音又は人声音に近づけることができる。

【0015】

【実施例】以下、この発明の実施例を添付図面に従って詳細に説明する。図1はこの発明に係る電子楽器の一実施例のハードウェア構成を示すブロック図である。この実施例において、楽音合成装置全体の制御は、マイクロプロセッサユニット（CPU）10と、システムプログラムや変更の必要のない各パラメータ等を格納するプロ

5

グラムROM11と、各種データを一時的に格納し、ワーキング用RAMとして用いられるデータ及びワーキングRAM12を含むマイクロコンピュータによって行われる。

【0016】プログラムROM11にはフォルマントシーケンスのプリセット値が格納され、またデータ及びワーキング用RAM12にはフォルマントシーケンス、ピッチシーケンス及び有聲無聲音レベルシーケンスの配列を記憶する領域が割り当てられている。従って、電源投入時のイニシャライズ処理で、プログラムROM11内のフォルマントシーケンスのプリセット値がロードされるので、楽音合成装置は電源投入時からフォルマントシーケンスによる楽音合成を行う。

【0017】このマイクロコンピュータには、データ及びアドレスバス13を介して、鍵盤回路14、操作パネル15、音源となる有聲フォルマント合成音源17V及び無聲フォルマント合成音源17U、シーケンスメモリ20、アフタタッチセンサ21及びタイマ22等の各種装置が接続されており、これらの各装置はマイクロコンピュータによってそれぞれ制御される。

【0018】鍵盤回路14は、発生すべき楽音の音高を指定する鍵盤のそれぞれの鍵に対応して設けられた複数のキースイッチからなる回路を含む。マイクロコンピュータにより、この鍵盤回路14の出力に基づき押圧鍵を複数の発音チャンネルのいずれかに割り当てるための発音処理が行われる。また、必要に応じて押し下げ時の押鍵操作速度を判別してイニシャルタッチデータITDを生成する処理も行われる。また、鍵盤の各鍵に関連して、鍵押圧持続時における押圧力を検出してアフタタッチデータATDを出力するアフタタッチセンサ21が鍵盤回路14に隣接して設けられている。

【0019】操作パネル15は、音色、音量、音高、効果等を選択・設定・制御するための各種操作子を含むものであり、ピアノ、オルガン、バイオリン、金管楽器、ギター等の各種自然楽器に対応する音色やその他の音色（人声音）を選択するための音色選択部16を有する。この音色選択部16は音色選択信号TCを出力する。

【0020】有聲フォルマント合成音源17Vは、複数のnチャンネルで楽音信号の同時発生が可能であり、データ及びアドレスバス13を経由して与えられる各チャンネルに割り当てられた鍵のキーコードKC、フォルマントパラメータデータ（中心周波数データF及びそのレベルデータL）、相対ピッチデータP、有聲音レベルデータV（又は有聲無聲音のレベル比UV）及びその他のデータ（キーオンKON、キーオフKOF、イニシャルタッチデータITD、アフタタッチデータATD、音色選択信号TC等）を入力し、これらの各種データに基づき楽音信号を発生する。

【0021】すなわち、フォルマント合成音源17Vは

(4)

6

特開平4-251297

シーケンスメモリ20から読み出された各フォルマントの中心周波数データF、レベルデータL、相対ピッチデータP、有聲音レベルデータV（レベル比UV）、合成すべき楽音の音高を指定するデータとして鍵盤回路14から与えられるキーコードKC等を入力し、所定のフォルマント合成演算を行い、そのフォルマントで特徴付けられる楽音信号をキーコードKC及び相対ピッチデータPで指定された音高に対応して、有聲音レベルデータV（レベル比UV）に応じたレベルで出力する。楽音の音高はキーコードKCに対応するピッチに相対ピッチデータPの値を乗算することによって得る。乗算以外の方法で音高を求めてもよいことはいままでもない。この実施例では有聲フォルマント合成音源17Vが同時発音可能なチャンネル数は8個として説明する。

【0022】無聲フォルマント合成音源17Uは、複数のnチャンネルで楽音信号の同時発生が可能であり、データ及びアドレスバス13を経由して与えられるフォルマントパラメータデータ（中心周波数データF及びそのレベルデータL）、無聲音レベルデータU（又は1-UV）及びその他のデータ（音色選択信号TC等）を入力し、これらの各種データに基づき楽音信号を発生する。

【0023】すなわち、フォルマント合成音源17Uはシーケンスメモリ20から読み出された各フォルマントの中心周波数データF、レベルデータL、無聲音レベルデータU（又は1-UV）を入力し、所定のフォルマント合成演算を行い、そのフォルマントで特徴付けられる楽音信号を無聲音レベルデータU（又は1-UV）に応じたレベルで出力する。この実施例では無聲フォルマント合成音源17Uが同時発音可能なチャンネル数は、有聲フォルマント合成音源17Vと同じ8個として説明する。

【0024】有聲フォルマント合成音源17V及び無聲フォルマント合成音源17Uから発生されたデジタル楽音信号はデジタル/アナログ（D/A）変換器18V、18Uによって、アナログの楽音信号に変換され、サウンドシステム19に出力される。サウンドシステム19はスピーカ及び増幅器等で構成され、D/A変換器18V、18Uからのアナログの楽音信号に応じた楽音を発生する。

【0025】タイマ22はマイクロコンピュータに対してインタラプト信号を定期的に与えるものである。この実施例では後述するフォルマントパラメータデータ、相対ピッチデータ及び有聲無聲音レベルデータの読み出し処理がタイマインタラプトによって実行される。シーケンスメモリ20は、複数のフォルマントに関する中心周波数データFやレベルデータL等の種々のフォルマントパラメータデータや相対ピッチデータ及び有聲無聲音レベルデータを所望のフォルマント変態態様に対応するシーケンスで記憶してなるものである。

【0026】図2はシーケンスメモリ20に記憶される

7

いるフォルマントパラメータデータ（中心周波数データFやレベルデータL）、相対ピッチデータP及び有声音無声音レベルデータV、Jの格納状態（メモリ構成例）を示す図である。図2において、縦軸は例えば人声音の音素等にそれぞれ対応しており、シーケンス番号X（1～m）で特定される。横軸はこの縦軸のシーケンス番号Xで特定された音素を特徴付けるフォルマントの中心周波数及びレベル、ピッチ、有声音レベル並びに無声音レベルの時系列的な変化に対応しており、ステップ番号Y（1～n）の順番に格納されている。

【0027】そして、シーケンスメモリ20の1個のステップには、シーケンス番号Xの音素を特徴付けるための中心周波数データFとレベルデータLからなる4個のフォルマントパラメータデータFLXYN（「X」はシーケンス番号、「Y」はステップ番号、「N」は1ステップ内の4個のうちの任意のものを示す番号）が格納されており、また、フォルマントパラメータデータFLXYNに対応して、そのフォルマント発音時の基準ピッチ（キーコードKCで特定されるもの）に対するピッチの相対的变化量を示す相対ピッチデータPXYと、そのフォルマント発音時の有声音のレベルを示す有声音レベルデータVXYと、無声音のレベルを示す無声音レベルデータUXY（「X」はシーケンス番号、「Y」はステップ番号を示す）がそれぞれ1個ずつ格納されている。この場合、各シーケンス番号Xにおけるステップ番号Yの数nの値は必ずしも同じでなくてもよく、各音素毎に異なっている。

【0028】また、シーケンスメモリ20は有声音レベルデータVXY及び無声音レベルデータUXYの代わりに有声音無声音レベル比データUVを記憶してもよい。この有声音無声音レベル比データUVは有声音レベルと無声音レベルの合計値を1とした場合における有声音の比率を示すものである。従って、無声音のレベルは1からこのレベル比UVを減算した値（1-UV）を演算することによって容易に求めることができる。シーケンスメモリ20に有声音無声音レベル比データUVを記憶することによって、シーケンスメモリ20のデータ量を大幅に削減できるという利点がある。

【0029】シーケンスメモリ20のシーケンス番号1のステップ番号1から読み出されるデータは、中心周波数データFとレベルデータLからそれぞれ構成される4個のフォルマントパラメータデータFL111、FL112、FL113、FL114と、相対ピッチデータP11と、有声音レベルデータV11と、無声音レベルデータU11とであり、ステップ番号nから読み出されるのは、4個のフォルマントデータFL1n1、FL1n2、FL1n3、FL1n4と、1個の相対ピッチデータP1nと、有声音レベルデータV1nと、無声音レベルデータU1nである。そして、これらのデータがステップ番号Yの順番に読み出される。

(5)

特開平4-251297

8

【0030】1個のステップに格納されたフォルマントパラメータデータは、バラレルに読み出されるようになっていてもよいし、時分割的に読み出されるようになっていてもよい。そして、ステップ順に読み出されるフォルマントパラメータデータFL111、FL121、・・・、FL1n1が時系列的に変化するフォルマントシーケンスデータであり、それぞれの中心周波数及びレベルが時系列的に微妙に変化している。

【0031】また、ステップ順に読み出される相対ピッチデータP11、P12、・・・、P1nが時系列的に変化するピッチシーケンスデータであり、基準ピッチに対するピッチの変化量を示すデータである。ステップ順に読み出される有声音レベルデータV11、V12、・・・、V1n及び無声音レベルデータU11、U12、・・・、U1nが時系列的に変化する有声音無声音レベルシーケンスデータである。各シーケンス番号のフォルマントパラメータデータ、相対ピッチデータ及び有声音無声音レベルデータも同様に構成されている。

【0032】シーケンスメモリ20は、シーケンス番号アドレスデータXとステップアドレスデータYとによって読み出し制御される。シーケンス番号アドレスデータXは、シーケンスメモリ20から読み出すべき一連のシーケンスフォルマントデータ及びシーケンスピッチデータのシーケンス番号を指定するものである。ステップアドレスデータYは、シーケンス番号アドレスデータXによって指定されたシーケンスにおいて現在読み出すべきステップを指定するものである。一例として、シーケンス番号アドレスデータXは、操作パネル15における適宜の選択操作に応じて、例えば音色選択部16に連動して、あるいは専用のシーケンス番号選択操作手段に応じて、発生される。また、ステップアドレスデータYは、CPU10によって発生され、シーケンスメモリ20にアドレス入力される。このステップアドレスデータの発生については後述する。

【0033】なお、この実施例では、一例として、フォルマントパラメータデータは、中心周波数データFとレベルデータLの2つであるとし、1つの楽音信号は4つのフォルマントによって有声音フォルマント合成音源17V及び無声音フォルマント合成音源17Uで合成されるものとして説明する。

【0034】シーケンスメモリ20に格納されるフォルマントパラメータデータは、従来記述の音声波形分析法を用いて生成する。例えば、線形予測分析法（LPC）、線スペクトル対分析法（LSP）又は複合正弦波モデル分析法（CSM）等の方法によって抽出されたフォルマントデータを用いる。

【0035】フォルマントデータの抽出と同時に相対ピッチデータの抽出も行う。相対ピッチの抽出は、変形自己相関法又はゼロクロス法等によって行う。即ち、原音声を採用する時、ある基準ピッチを定めておき、その近

9

傍のピッチにて発声してもらう。これに、ピッチのゆらぎ成分が含まれている。そこで、基準ピッチと分析ピッチとの差分をとり、これを相対ピッチデータとして抽出する。有聲無聲音レベルの抽出も相対ピッチデータの抽出と同様に行う。また、レベル比UVはフーリエ分析により約5 KHz以上とそれ以下のパワー比を求めることによって抽出できる。

【0036】操作パネル15には、複数の読出しパターンの中から1つの読出しパターンを選択するためのモード切換えキー（図示せず）を有する。CPU10はモード切換えキーによって選択された読出しパターンに従って複数ステップに関するフォルマントパラメータデータ、相対ピッチデータ及び有聲無聲音レベルデータを読み出すための時間的に変化するステップアドレスデータを発生する。この読出しパターンに従うステップアドレスデータの発生は、鍵盤回路14からのキーオン信号KONに応じて、楽音発音タイミングに対応して行われる。複数の読出しパターンに対応してステップアドレスデータはデータ及びワーキングRAM12内に予め記憶しておいてもよいし、あるいは演算により発生するようにしてもよい。

【0037】図3はシーケンスメモリ20に格納される有聲音レベルデータV、無聲音レベルデータU及び有聲音のレベル比UVの値の一例を模式的に示す図である。図3(a)は有聲音レベルデータVを示し、図3(b)は無聲音レベルデータUを示し、図3(c)は有聲音無聲音レベル比UVを示す。図3(a)～(c)の横軸はシーケンスメモリ20のステップアドレスを示し、縦軸はそのステップアドレスに対応した有聲音無聲音のレベル値を示す。

【0038】図3の有聲音及び無聲音のレベルは、それぞれ『あ、か、さ、た』（[a], [ka], [sa], [ta]）の人の声を楽音合成装置で合成発音する場合に対応している。図3(a)の各波形は全て有聲音（母音）[a]のレベルVaを示し、図3(b)の各波形の第1番目は無聲音（子音）[k]のレベルUkを、第2番目は無聲音（子音）[s]のレベルUsを、第3番目は無聲音（子音）[t]のレベルUtをそれぞれ示す。図3(a)において有聲音レベルVaの大きさは一定であるが、図3(b)においては無聲音レベルUk, Us, Utの大きさはそれぞれ異なっている。従って、楽音合成装置は図3(a)及び(b)の波形をステップアドレスの順に次々と読み出すことによって、『あ、か、さ、た』（[a], [ka], [sa], [ta]）の人の声を発音することができる。

【0039】図3(c)は、シーケンスメモリ20に格納される有聲音無聲音のレベル比UVを示すものである。この例では、有聲音レベルと無聲音レベルの合計値を1とした場合に、有聲音レベルがその合計に占める割合が有聲音無聲音レベル比UVとしてシーケンスメモリ20に

(6)

特開平4-251297

10

格納されている。即ち、有聲音レベルをV、無聲音レベルをUとした場合の $V/(U+V)$ が有聲音無聲音レベル比UVである。従って、無聲音レベルの大きさは $1-U/V$ （図では点線で示してある）を演算することによって容易に求めることができる。但し、図3(a)及び(b)に示すように、有聲音レベルと無聲音レベルの合計値が1でない場合には、有聲音レベルVと無聲音レベルUとを別々にシーケンスメモリ20に格納する方がよいことは言うまでもない。また、有聲音レベルと無聲音レベルの合計が1でない場合でも、単純に両者の比率 (V/U) を格納してもよいことは言うまでもない。

【0040】図4はシーケンスメモリ20の読出しパターンの幾つかの例を示す図である。図4(a)に示す読出しパターンは、所定の基準アドレスから所定の最大アドレス(MAX)まで、ステップアドレスを順次増加させ、その後は最大アドレス(MAX)を維持するものである。例えば、基準アドレスをステップ番号1、最大アドレスをステップ番号nとすると、発音開始後、ステップ番号1, 2, 3……nの順で時間的にステップアドレスが増加し、ステップ番号nに到達すると、以後はステップ番号nを維持する。この場合、例えば、シーケンス番号1の場合は、各ステップ番号1, 2, 3……nに対応するフォルマントパラメータデータFL11N, FL12N, ……FL1nN、相対ピッチデータP11, P12, ……P1n、有聲音レベルデータV11, V12, ……V1n及び無聲音レベルデータU11, U12, ……U1nが順次読み出され、最後にFL1nN, P1n, V1n及びU1nが持続的に読み出される。なお、これ以後の説明の中でアドレスという場合は、ステップアドレスを指すものとする。

【0041】図4(b)の読み出しパターンは所定の基準アドレスから所定のループエンドアドレス(LOOP END)まで所定の順序で読み出し、その後はループスタートアドレス(LOOP START)からループエンドアドレス(LOOP END)に向かってアドレスを増加するというパターンを繰り返すものである。

【0042】図4(c)の読み出しパターンは所定の基準アドレスから所定のループエンドアドレス(LOOP END)まで所定の順序で読み出し、その後はループエンドアドレス(LOOP END)からループスタートアドレス(LOOP START)に向かってアドレスを減少するというパターンを繰り返すものである。

【0043】図4(d)の読み出しパターンは所定の基準アドレスから所定のループエンドアドレス(LOOP END)まで所定の順序で読み出し、その後はループエンドアドレス(LOOP END)とループスタートアドレス(LOOP START)との間をアドレスが増加減少するように繰り返すものである。

【0044】図4に示した読み出しパターンは一例であり、これらのパターンを任意に組み合わせることもでき

11

るし、読み出し速度をインシタルタッチデータITD、アフタタッチデータATDの大きさ等のキー操作情報に応じて可変制御してもよい。また、楽器の種類、人声の種類に応じて種々のパターンを予め登録しておくことによって、演奏効果を格段に向上することができる。

【0045】なお、操作パネル15にマニュアル操作子を設け、このマニュアル操作子で任意のステップアドレスを設定し、また操作量を変化することにより順次変化させることもできる。このマニュアル操作子として、例えば128段階の信号を出力するモジュレーションホイールや正負の方向性を持った信号を出力するピッチベンドホイール等がある。

【0046】次に、マイクロコンピュータによって実行される処理の一例を図5、図6、図7、図8及び図9に基づいて説明する。図5はマイクロコンピュータが処理する「メイン処理ルーチン」の各ステップの処理内容を示す図である。このメイン処理ルーチンは次のようなステップで順番に実行される。

【0047】ステップ31：電源投入時におけるマイクロコンピュータの全てのデータに所定の値をセットする。例えば、シーケンス番号アドレスデータ、ステップアドレスデータ及び読み出しパターン等の初期値を各レジスタに設定する。また、前述のようにプログラムROM11に格納されているフォルマントシーケンスのプリセット値をデータ及びワーキング用RAM12にロードする。

【0048】ステップ32：鍵盤回路14における各キースイッチのスキャンを行う。

ステップ33：ステップ32のキースキャンの結果、キーイベントの有無を判断する。鍵が押し下げられたときはキーオンイベントが判断され、離鍵されたときはキーオフイベントが判断される。キーイベント有り(YES)の場合は、次のステップ34に進み、キーイベント無し(NO)の場合はステップ35に進む。

【0049】ステップ34：ステップ33のキーイベントの種類に応じた発音割当て処理を行う。

ステップ35：操作パネル15の各操作子のスキャンを行う。

ステップ36：ステップ35のパネルスキャンの結果、操作子によるパネルイベントの有無を判断する。例えば、モード切換えキーが押されているかどうかを判断する。パネルイベント有り(YES)の場合は、次のステップ37に進み、パネルイベント無し(NO)の場合は、ステップ32に戻る。

ステップ37：ステップ36のパネルイベントの結果に応じて発音割当て処理を行う。

【0050】次に、図6はマイクロコンピュータが処理する「発音処理」の各ステップの処理内容を示す図である。この発音処理は次のようなステップで順番に実行される。

(7)

特開平4-251297

12

ステップ41：キーイベントがキーオン又はキーオフのどちらなのかを判断する。キーイベントがキーオン(YES)の場合は次のステップ42に進み、キーオフ(NO)の場合はステップ47に進む。

ステップ42：有声フォルマント合成音源17V及び無声フォルマント合成音源17Uのそれぞれが発音割当て処理を行っていない空きチャンネルをサーチする。

【0051】ステップ43：ステップ42の空きチャンネルサーチの結果、空きチャンネルの有無を判断し、空きチャンネル有り(YES)の場合はステップ45にジャンプし、空きチャンネル無し(NO)の場合は次のステップ44に進む。

ステップ44：空きチャンネルが無かったので、トランケート処理を行い、発音割当て処理可能なチャンネルを作成する。

【0052】ステップ45：インシタルタッチデータITDに応じたステップアドレスの読み出し位相のスピードを決定し、これをスピードレジスタSP(ch)に設定する。このスピードレジスタSP(ch)はチャンネル毎に設けられており、チャンネル毎に異なった値が設定される。従って、押鍵毎にフォルマントシーケンス、ピッチシーケンス、有声無声レベルシーケンスの読み出し速度が変化するので、和音等が押鍵された場合でも各音間の分離性が向上するという効果がある。

【0053】ステップ46：当該空きチャンネルにキーオンフラグをセットして、割り込み処理における発音情報の制御が行われるようにし、リターンする。すなわち、この発音処理では直接には発音割当て要求は行わない。

ステップ47：ステップ41の結果、キーイベントがキーオフであった場合、対応するキーの発音処理を有声フォルマント合成音源17V及び無声フォルマント合成音源17Uが行っているかどうか、すなわち対応するキーオンが存在するかどうかを判断し、存在する(YES)の場合は次のステップ48に進み、存在しない場合はリターンする。これは、キーオン処理されたキーがステップ44のトランケート処理によってその発音が完全に停止していることがあり得るからである。

【0054】ステップ48：キーオフイベントの該当するチャンネルに対してキーオフ情報KOFを送出する。

ステップ49：キーオフ情報KOFの送出されたチャンネルのキーオンフラグをリセットする。

【0055】図7はマイクロコンピュータが処理する「パネル処理」の各ステップの処理内容を示す図である。このパネル処理は次のようなステップで順番に実行される。ここでは、この発明に直接関係する部分のみが示されている。

ステップ51：操作パネル15上のモード切換えキーが押されたかどうかを判断し、押された(YES)場合は次のステップ52に進み、押されていない(NO)場合

13

はステップ56に進む。

【0056】ステップ52: モードレジスタMODEの値を「1」だけインクリメントし、次のステップ53に進む。すなわち、このモード切換えスイッチは一回押すたびにモードレジスタMODEの値を増加させ、図4の読み出しパターンの中から任意のものを選択するものである。モード切換えキーはこれに限らずスライド又は回転によってその値を任意に変えられるようなものであってもよいことはいうまでもない。

【0057】ステップ53: 図4の読み出しパターンは4種類なので、モードレジスタMODEの値がモード数の最大値「5」であるかどうかを判断し、「5」の場合は次のステップ54に進み、そうでない場合はステップ55にジャンプする。読み出しパターンの数に応じてこのステップの値を変更してやればよい。

【0058】ステップ54: ステップ53でモードレジスタMODEの値が最大値「5」だったので、ここでモードレジスタMODEに「1」を格納する。このようにしてモードレジスタMODEの値はオーバーフローすることなく、モード切換えスイッチの押圧動作によって1, 2, 3, 4の値を循環する。

【0059】ステップ55: モードレジスタMODEに格納されている値に応じて割り込み処理ベクタを書き換える。この実施例では、フォルマントシーケンスデータ、ピッチシーケンスデータ及び有聲無聲音レベルシーケンスデータの読み出し波形をソフトウェア処理で作り出しており、更にそれぞれの読み出し波形を異なる割り込み処理で形成するようにしているので、モードレジスタMODEの格納値に応じて割り込み処理先を変更している。

【0060】ステップ56: 操作パネル15上のモード切換えキー（図示せず）が押されていない場合、操作パネル15上のその他のスイッチキーに対する処理を行う。例えば、音色選択イベントが発生した場合等は、その音色に応じてシーケンス番号アドレスを変更したりする。

【0061】図8はマイクロコンピュータが処理するフォルマントパラメータデータ、相対ピッチデータ、有聲無聲音レベルデータの読み出し処理の詳細を示す図である。図8に示した処理は図7のモードレジスタMODEの格納値が「1」の場合、すなわち図4(a)の読み出しパターンにおける割り込み処理を示すものである。このルーチンはタイマ22からインタラプト信号が与えられる毎に次のようなステップの順番に実行される。

【0062】ステップ61: 割り込みの多重化を防止するために割り込み禁止を行う。

ステップ62: チャンネルナンバレジスタchに「1」を設定する。

ステップ63: チャンネルナンバレジスタchに格納さ

(8)

特開平4-251297

14

れている値に対応するチャンネルのキーオンフラグがセットされているかどうかを判断し、セットされている（YES）場合は次のステップ64に進み、セットされていない（NO）場合はステップ613にジャンプし、チャンネルナンバレジスタchの値を1だけインクリメントして、次のチャンネルの処理を行う。

【0063】ステップ64: ステップアドレスレジスタY(ch)の値を「1」だけインクリメントし、さらにスピードレジスタSP(ch)にセンシティブィーSEを乗じた値 $SP(ch) \times SE$ を加算する。このセンシティブィーSEを「0」にすると、スピードレジスタSP(ch)の値には無関係にステップ番号Yは1ずつ増加し、センシティブィーSEを正の値にすると、ステップ番号Yの増加率は大きくなり、逆にセンシティブィーSEを負の値にすると、ステップ番号Yの増加率は小さくなる。

【0064】すなわち、センシティブィーSEが「0」だと、シーケンスメモリ20からはFL11N, FL12N, FL13N, ...のようにステップ番号の順番通りにフォルマントパラメータデータ及び相対ピッチデータが読み出される。センシティブィーSEが正の値だと、シーケンスメモリ20からは例えばFL11N, FL13N, FL15N, ...のように飛び飛びのステップ番号順に読み出されフォルマントパラメータデータ、相対ピッチデータ又は/及び有聲無聲音レベルデータの読み出し速度が早くなる。逆に負の値だと、例えばFL11N, FL11N, FL12N, FL12N, FL13N, FL13N, ...のように同じステップ番号が読み出されフォルマントパラメータデータ、相対ピッチデータ又は/及び有聲無聲音レベルデータの読み出し速度が遅くなる。

【0065】なお、 $SP(ch) \times SE$ の代わりに、スピードレジスタSP(ch)だけを使用し、このスピードレジスタSP(ch)にタッチデータに応じた正負の値を格納できるようにしてもよいし、センシティブィーSEの値及び/又はスピードレジスタSP(ch)の値をイニシャルタッチデータITD及び/又はアフタタッチデータATDの値に応じて可変制御してもよい。

【0066】ステップ65: ステップアドレスレジスタY(ch)がそのシーケンス番号における最大ステップ値MAXよりも大きいかどうかを判断し、大きい場合はステップ66に進み、小さい場合はステップ67に進む。

ステップ66: シーケンス番号の最大ステップ値MAXよりも大きいステップ番号は存在しないので、ステップアドレスレジスタY(ch)にそのシーケンス番号の最大ステップ値MAXを格納する。

【0067】ステップ67: 各ステップ番号Yの4個のフォルマントパラメータデータを順次読み出すためにレジスタNに「1」を格納する。

15

ステップ68:シーケンスメモリ20からシーケンス番号X及びステップ番号Yに対応する有声音レベルデータVXY及び無声音レベルデータUXYを読み出し、それぞれ有声音フォルマント合成音源17V及び無声音フォルマント合成音源17Uに出力する。これによって、有声音及び無声音の微妙なレベル変化を表現することができる。なお、シーケンスメモリ20に図3(c)の有声音のレベル比UVが格納されている場合には、そのレベル比UVを有声音フォルマント合成音源17Vに出力し、有声音のレベル比1-UVを演算し、それを無声音フォルマント合成音源17Uに出力する。

【0068】ステップ69:シーケンスメモリ20からシーケンス番号X及びステップ番号Yに対応する相対ピッチデータPXYを読み出し、キーコードKCと共に有声音フォルマント合成音源17Vに出力する。これによって、微妙な音程のズレを表現できる。なお、相対ピッチデータPXY及びキーコードKCは無声音フォルマント合成音源17Uには取り込まれない。

ステップ610:シーケンスメモリ20からシーケンス番号X及びステップ番号Yに対応したフォルマントパラメータデータFLXY1を読み出し、有声音フォルマント合成音源17V及び無声音フォルマント合成音源17Uに出力する。

ステップ611:レジスタNを「1」だけインクリメントする。

【0069】ステップ612:レジスタNが「4」よりも大きいかどうか判断し、大きい場合はステップ番号Yの4個のフォルマントパラメータデータFLXY1~FLXY4を読み出し、小さい場合はまだ4個のフォルマントパラメータデータを読み出していないので、ステップ610に戻り、4個のフォルマントパラメータデータを読み出すまで処理を実行する。

【0070】ステップ613:チャンネルナンバレジスタchを「1」だけインクリメントする。

ステップ614:チャンネルナンバレジスタchが「8」よりも大きいかどうか判断し、大きい場合は次のステップ615に進み、小さい場合はステップ63に戻り次のチャンネルに対して同様の処理を実行する。

ステップ615:ステップ61で禁止していた割り込みを許可し、通常の処理にリターンする。

【0071】図9は図8と同様、マイクログンピュータが処理するフォルマントパラメータデータ、相対ピッチデータ及び有声音無声音レベルデータの読み出し処理の詳細を示す図であり、図7のモードレジスタMODEの格納値が「2」の場合、すなわち図4(b)の読み出しパターンにおける割り込み処理を示すものである。ステップ71~74、77~715は図8のステップ61~64、67~615と同じなので、その説明は省略する。

16

特開平4-251297

【0072】ステップ75:ステップアドレスレジスタY(ch)がそのシーケンス番号におけるループエンドアドレス値Lよりも大きいかどうかを判断し、大きい場合はステップ76に進み、小さい場合はステップ77に進む。

ステップ76:ステップアドレスレジスタY(ch)にそのシーケンス番号におけるループスタートアドレス値LSを格納する。

【0073】このステップ75、76によって、図4(b)のような読み出しパターンに従って、フォルマントパラメータデータ、相対ピッチデータ及び有声音無声音レベルデータをシーケンスメモリ20から順次読み出すことができる。なお、図4(c)及び(d)の読み出しパターンは図8及び図9のフローを変形することによって容易に実現できるので、ここでは省略する。

【0074】以上のようにこの実施例によれば、フォルマントを時系列的に変化させて発生することができると共に楽音のピッチ及び有声音無声音レベルも時系列的に変化させることができる。また、そのときに読み出し速度を変更することによって、フォルマント、ピッチ及び有声音無声音レベルの変化の割合を自由に制御できるという効果がある。

【0075】なお、有声音フォルマント合成音源17Vにおける楽音合成方式にはいかなる方式のものを用いてもよい。例えば、特公昭59-19352号公報又は特公昭62-14834号公報に記載されているような振幅変調(AM)方式、又は特公昭62-42515号公報に記載されているような周波数変調(FM)方式などを使用することができる。この有声音フォルマント合成音源17Vの一例として窓関数を用いた振幅変調(AM)方式により、有声音フォルマントを合成するものを図10に示す。

【0076】図10において、位相発生器81は、フォルマント中心周波数を設定するための中心周波数データFを逐次累算することによりフォルマント中心周波数に対応する位相データを発生するものである。従って、フォルマント中心周波数の値が小さい時は累算の速度は遅く、フォルマント中心周波数が大きい時は累算の速度は速くなる。そして、累算値がオーバーフローしたときは、再び初期値に戻って累算を繰り返す。また、所定時間幅のリセットパルスRSが与えられたときは、累算値が0にリセットされ、該所定時間幅の間だけ出力0を維持する。この位相発生器81の累算出力はセクタ85を介して対数表現の正弦関数($\log \sin$)テーブル86にアドレスデータとして供給される。

【0077】位相発生器82は、アキュムレータ構成であり、キーコードKCに対応する基本ピッチ周波数データf0を取り込み、この基本ピッチ周波数データf0を順次累算する。この位相発生器82もオーバーフローすると初期値に戻って累算動作を繰り返す。また、位相発

17

生器82はオーバーフローパルス(例えば、最上位ビットMSB)を微分回路83に出力するようになっている。

【0078】微分回路83は、ワンショットマルチバイブレータから構成され、オーバーフローパルスが立ち上がると、所定時間幅のリセットパルス信号RSを位相発生器81及び84に出力する。すなわち、微分回路83は、位相発生器82の出力値が0になるタイミングを検出し、そのタイミングでリセットパルス信号RSを出力する。従って、位相発生器81で発生するフォルマント中心周波数の位相データは、このリセットパルス信号RSに応じて発生すべき楽音の音高に応じた周期で所定時間幅の間リセットされ、これによりフォルマント中心周波数を搬送周波数とし、楽音の音高周波数を変調周波数とする振幅変調が行われる。

【0079】位相発生器84は、図示していない音色パラメータ供給回路から供給される音楽変調位相定数Kを所定クロックに同期して累算する回路である。この位相発生器84は、累算値がオーバーフローすると、その最終値を保持し、次いでリセットパルス信号RSが供給されると、その内容をリセットし、再び初期値から累算を開始するようになっている。この位相発生器84の累算結果は、セレクト85を介して対数表現の正弦関数($\log \sin$)テーブル86にアドレスデータとして供給される。この場合、位相発生器84の累算速度は、位相発生器81の累算速度に比べて極めて遅くなるように音楽変調位相定数Kの値が設定されている。

【0080】セレクト85は動作選択信号SELが供給されたときに位相発生器81の出力データを選択し、動作選択信号SELが供給されていないときに位相発生器84の出力データを選択し、正弦関数テーブル86にアドレスデータとして供給する。

【0081】正弦関数テーブル86は、対数表現の正弦関数データが1周期分(又は1/2周期あるいは1/4周期分でもよい)記憶されているテーブルであり、セレクト85を介して供給されるアドレスデータに応じた正弦関数値を対数表現で出力するようになっている。従って、正弦関数テーブル86は、位相発生器81又は84における累算値に応じたレートで正弦関数値を出力する。

【0082】データシフト87は、正弦関数テーブル86の出力データを、音色パラメータであるシフト量データSに従ってシフトする回路である。このシフト量データSも図示しない音色パラメータ供給回路から供給される。このデータシフト87は、動作信号SFTが供給されているときにシフト動作を行い、動作信号SFTが供給されていないときは正弦関数テーブル86からのデータがそのまま出力する。また、データシフト87におけるシフトは、シフト量データSの値だけ上位側にシフトする動作となる。

(10)

特開平4-251297

18

【0083】加算器88は、動作信号ADD1が供給されると、データシフト87の出力データとレジスタ89の出力データとを加算する。動作信号ADD1が供給されていない場合は、加算器88に供給されるデータは、そのまま出力端から出力される。また、レジスタ89には、加算器88をそのまま通過したデータが記憶されるようになっている。この場合、加算器88による加算は、対数データについての加算であるから、真数に対しては乗算を行うことになる。

【0084】加算器810は、動作信号ADD2が供給されると、加算器88の出力データとレベル変換されたレベルデータLとを加算する。この加算器810における加算は、対数値の加算であるから真数に対しては乗算を行っていることとなる。

【0085】対数/リニア(\log linear)変換回路811は、加算器810から供給される対数表現のデータを真数に変換する回路である。対数/リニア変換回路811が出力するデータは、アキュムレータ812に与えられる。1つの楽音信号を合成するための4つのフォルマントパラメータデータF、Lが時分割的に与えられ、各フォルマントに応じた部分音信号が対数/リニア変換回路811から順次出力され、これがアキュムレータ812で累算され、楽音信号として乗算器813に出力される。乗算器813は有声音レベルデータV(又はUV)を入力し、それをアキュムレータ812からの楽音信号に乗算して、有声音の楽音信号として出力する。

【0086】以上が有声音フォルマント合成音源17Vの一例であるが、この詳細については特願平1-77383号明細書に記載してあるので、ここでは省略する。なお、図10の有声音フォルマント合成音源17Vのうち、位相発生器84、セレクト85及びデータシフト87は省略してもよい。

【0087】図11は、無声音フォルマント合成音源17Uの一例を示す図である。この詳細については特願平2-271397号公報の明細書に記載してあるので、ここでは簡単に説明する。図11において、ホワイトノイズ発生回路91はフラットなスペクトルを持つホワイトノイズを発生する回路である。デジタルフィルタ92はIIRフィルタと呼ばれるローパスフィルタであり、フラットなスペクトルを持つホワイトノイズを所定のバンド幅を有するノイズに変換するものである。デジタルフィルタ92はインバータ93、バンド幅パラメータ発生器94、遅延回路95、加算器96、97、98及び乗算器99からなる。デジタルフィルタ92は、ホワイトノイズを右下がりスペクトル特性を有するノイズ信号に変換して乗算器910に出力する。

【0088】周期波形発生回路911はフォルマント中心周波数を設定するための中心周波数データFに対応して遷移するフォルマント中心周波数foを有する正弦波

19

の順次サンプル点振幅値 $\sin 2\pi f_0 t$ を出力する。周期波形発生回路911は位相アキュムレータ912及びサインテーブル913からなる。位相アキュムレータ912は所定のクロックパルスに同期して中心周波数データFを累算する。この中心周波数データFは発生しようとするノイズ音のフォルマント中心周波数 f_0 に対応しているため、位相アキュムレータ912はその累算値をサインテーブル913の読み出しアドレス信号として出力する。

【0089】サインテーブル913には、正弦関数データが1周期分（又は1/2周期あるいは1/4周期分でもよい）記憶されているテーブルであり、読み出しアドレス信号によって読み出される。従って、サインテーブル913からは位相アキュムレータ912の読み出しアドレス信号（累算値）に応じた周波数 f_0 の正弦波が出力する。乗算器910はデジタルフィルタ92のノイズ信号と周期波形発生回路911の正弦波とを乗算して出力する。従って、乗算器910からは所定のフォルマント特性を持ったノイズ信号が出力されるようになる。

【0090】エンベロープジェネレータ914はフォルマントパラメータのレベルデータLに従ってクロックパルスのタイミングで乗算器910から出力されるノイズ信号の振幅を制御するエンベロープ信号を乗算器915に出力する。乗算器915は乗算器910からのノイズ信号にこのエンベロープ信号を乗算して次段の乗算器916に出力する。乗算器916は無声音レベルデータU（又は1-UV）を入力し、それを乗算器915からのノイズ信号に乗算して、無声音の楽音信号として出力する。

【0091】なお、上述の実施例ではソフトウェアによって実施する場合について説明したが、これに限らずハードウェア等で実施するようにしてもよい。シーケンスメモリ20内のフォルマントパラメータデータの内容は操作パネル13の操作によって任意に書換え可能としてもよい。

【0092】上述の実施例では、図2のステップ番号を図4の読み出しパターンに従って読み出す場合について説明したが、これに限らず図4の読み出しパターンに応じてステップ番号からシーケンス番号にまたがってフォルマントパラメータデータを順次読み出すようにしてもよい。また、フォルマントパラメータデータ、相対ピッチデータ及び有声無声音レベルデータの全てを読み出す場合について説明したが、いずれか一つを読み出すだけでもよい。

【0093】また、フォルマントパラメータデータに比べてピッチ及び有声無声音レベルの変化量も少なく、変化速度も遅いので、ステップ番号毎に相対ピッチデータ及び有声無声音レベルを有しなくてもよく、数ステップ毎に相対ピッチデータ及び有声無声音レベルを設けるようにしてもよいし、フォルマントパラメータデータ、相

(11)

特開平4-251297

20

対ピッチデータ及び有声無声音レベルを別々のシーケンスメモリに記憶し、別々のシーケンススピードで読み出すようにしてもよいし、異なるステップ番号を与えるようにしてもよい。

【0094】相対ピッチデータは分析によって抽出してもよく、又は専用のエディター等により作成したピッチデータを用いてもよい。相対ピッチデータ又は有声無声音レベルデータの出力をオフ状態にするスイッチを設けてもよいし、相対ピッチデータ又は有声無声音レベルデータの深さ（大きさ）を増減するスイッチを設けてもよい。これはピッチ変化や有声無声音レベル変化の全くないフラットな楽音の発音が欲しい場合もあるからである。フォルマントパラメータデータもピッチデータと同様に基準フォルマントデータに対する相対変化量を時系列的に順次記憶するようにしてもよい。

【0095】また、実施例では、シーケンスメモリ20に記憶するフォルマントパラメータデータとして、フォルマントを特定するための中心周波数及びレベルのパラメータを記憶する場合について説明したが、周波数変調演算によってフォルマントを合成する場合はそのための中心周波数、変調周波数、変調指数及びレベル等の各パラメータを記憶しておき、これを読み出すようにしてもよい。また、音階音に限らず、リズム音等のフォルマントを合成する場合にも適用できることは言うまでもない。

【0096】

【発明の効果】この発明によれば、時系列的に変化するようなフォルマントパラメータデータ相対ピッチデータ又は有声無声音レベルデータを複数ステップにわたって予め記憶手段に記憶し、このフォルマントパラメータデータ、相対ピッチデータ又は有声無声音レベルデータを読み出し手段で複数ステップにわたって順次読み出すことによって、楽音信号のフォルマント、ピッチ又は有声無声音レベルを実際の楽器音又は人声音と同じように自然に変化させることができる。

【図面の簡単な説明】

【図1】 この発明に係る電子楽器の一実施例のハードウェア構成を示すブロック図である。

【図2】 図1のシーケンスメモリに記憶されているフォルマントパラメータデータ及び相対ピッチデータの状態を示す図である。

【図3】 シーケンスメモリに格納される有声音レベルデータ及び無声音レベルデータの値の一例を模式的に示す図である。

【図4】 図1のシーケンスメモリの読み出しパターンのいくつかの例を示す図である。

【図5】 図1のマイクロコンピュータが処理するメインルーチンの一例を示すフローチャート図である。

【図6】 図1のマイクロコンピュータが処理する図4の発音処理の詳細例を示すフローチャート図である。

【図7】 図1のマイクロコンピュータが処理する図4のパネル処理の詳細例を示すフローチャート図である

【図8】 図3(a)の読み出しパターンに従ったフォルマントパラメータデータ、相対ピッチデータ及び有声音レベルデータの読み出し処理の詳細例を示すフローチャート図である。

【図9】 図3(b)の読み出しパターンに従ったフォルマントパラメータデータ、相対ピッチデータ及び有声音レベルデータの読み出し処理の詳細例を示すフローチャート図である。

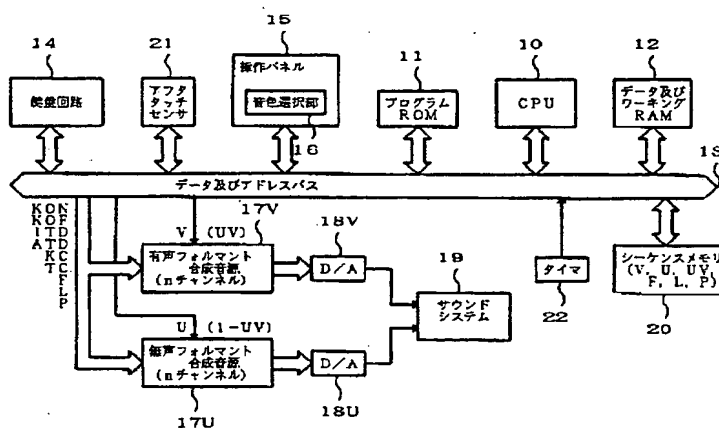
【図10】 図1の有声フォルマント合成音源の一例を示す図である。

【図11】 図1の無声フォルマント合成音源の一例を示す図である。

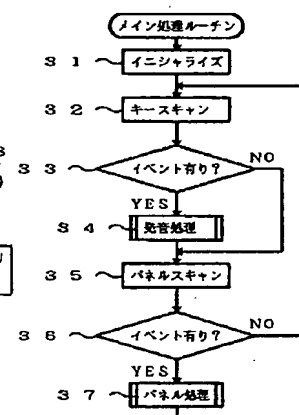
【符号の説明】

10…CPU、11…プログラムROM、12…データ及びワーキングRAM、13…データ及びアドレスバス、14…鍵盤回路、15…操作パネル、16…音色選択部、17V…有声音フォルマント合成音源、17U…無声音フォルマント合成音源、18…D/A変換器、19…サウンドシステム、20…シーケンスメモリ、21…アフタタッチセンサ、22…タイマ

【図1】



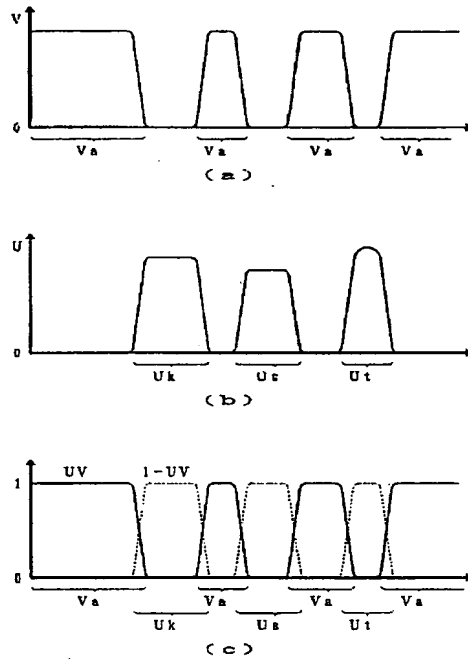
【図5】



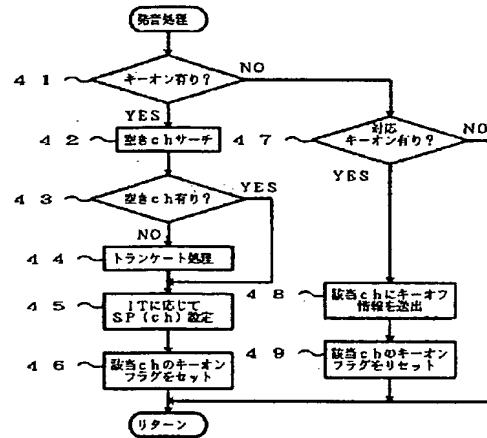
【図2】

		ステップ番号 Y																			
		1								2								n			
		1	2	3	4	P	V	U	1	2	3	4	P	...	4	P	V	U			
シー ケ ン ス 番 号	1	FL111	FL112	FL113	FL114	P11	V11	U11	FL121	FL122	FL123	FL124	P12	...	FL1n4	P1n	V1n	U1n			
	2	FL211	FL212	FL213	FL214	P21	V21	U21	FL221	FL222	FL223	FL224	P22	...	FL2n4	P2n	V2n	U2n			
	3	FL311	FL312	FL313	FL314	P31	V31	U31	FL321	FL322	FL323	FL324	P32	...	FL3n4	P3n	V3n	U3n			
	4	FL411	FL412	FL413	FL414	P41	V41	U41	FL421	FL422	FL423	FL424	P42	...	FL4n4	P4n	V4n	U4n			
	5	FL511	FL512	FL513	FL514	P51	V51	U51	FL521	FL522	FL523	FL524	P52	...	FL5n4	P5n	V5n	U5n			
				
X		FLx11	FLx12	FLx13	FLx14	Px1	Vx1	Ux1	FLx21	FLx22	FLx23	FLx24	Px2	...	FLxn4	Pxn	Vxn	Uxn			

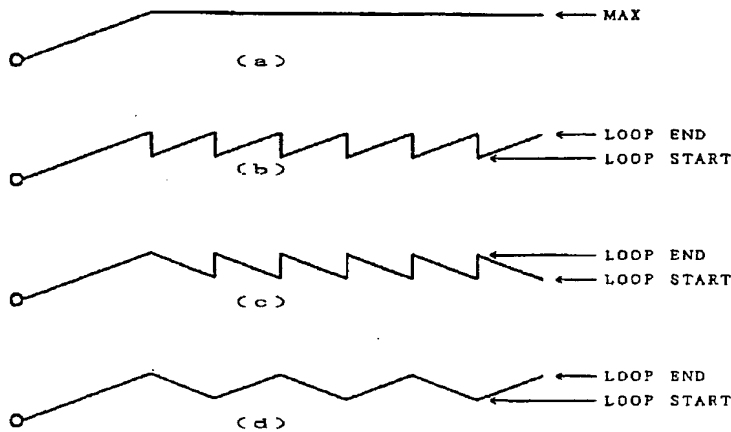
【図3】



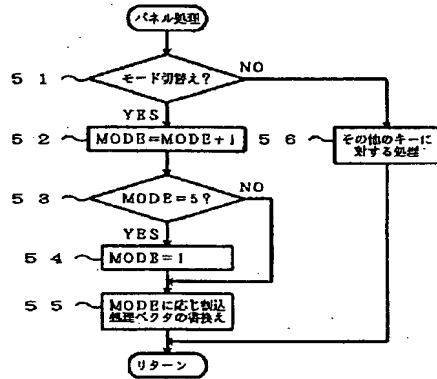
【図6】



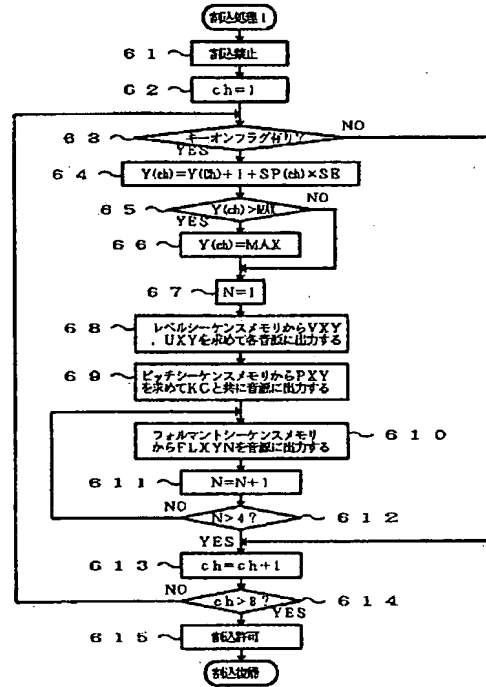
【図4】



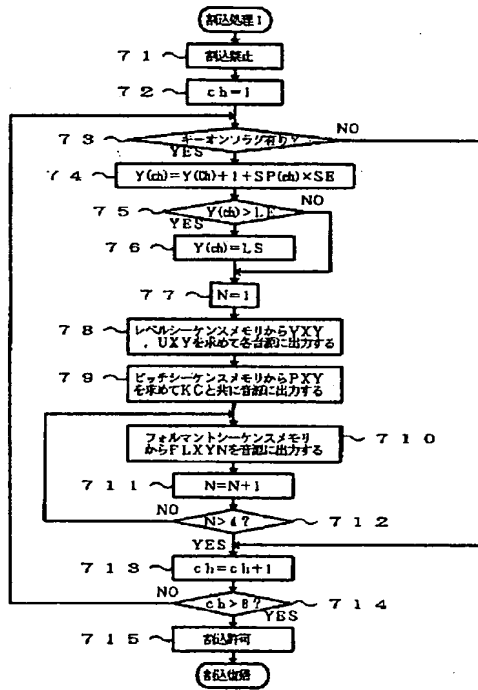
【図7】



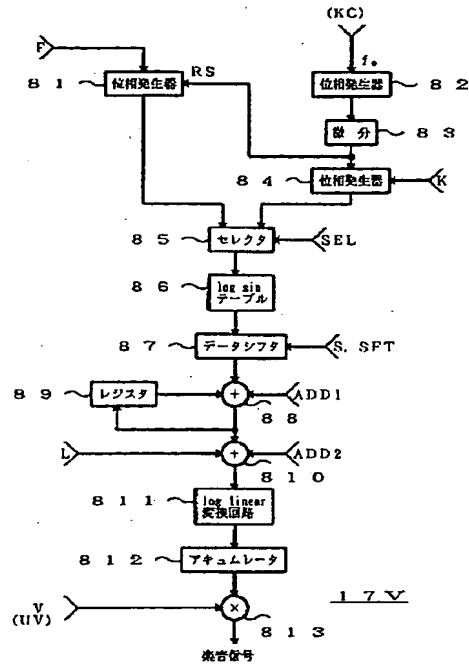
【図8】



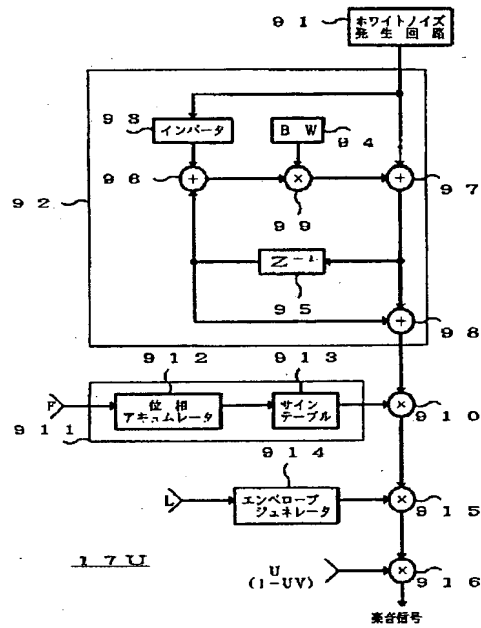
【図9】



【図10】



【図11】



**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☐ **FADED TEXT OR DRAWING**
- ☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.